

# Oracle® Fusion Middleware

## Installing and Configuring Oracle HTTP Server



14c (14.1.2.0.0)

F85506-01

December 2024

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Oracle Fusion Middleware Installing and Configuring Oracle HTTP Server, 14c (14.1.2.0.0)

F85506-01

Copyright © 2018, 2024, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

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

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

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

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

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

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

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

# Contents

## Preface

---

Audience	vi
Documentation Accessibility	vi
Diversity and Inclusion	vi
Related Documents	vi
Conventions	vii

## 1 About the Oracle HTTP Server Installation

---

About Oracle HTTP Server	1-1
Using the Standard Installation Topology As a Starting Point	1-2
About the Standard Installation Topology for Oracle HTTP Server in a WebLogic Server Domain	1-2
Understanding Elements in the Standard Installation Topology Illustration	1-3
About Full-JRF and Restricted-JRF Modes	1-4
About the Standard Installation Topology for Oracle HTTP Server in a Standalone Domain	1-5

## 2 Preparing to Install and Configure Oracle HTTP Server

---

Selecting an Oracle HTTP Server Installation Type	2-1
Roadmap for Installing and Configuring Oracle HTTP Server in a Collocated Domain	2-2
Roadmap for Installing and Configuring Oracle HTTP Server in a Standalone Domain	2-4
Roadmap for Verifying Your System Environment	2-4
Verifying Certification, System, and Interoperability Requirements	2-5
Selecting an Installation User	2-6
About User Permissions	2-6
About Non-Default User Permissions on Linux or UNIX Operating Systems	2-8
Verifying That the Installation User Has Administrator Privileges on Windows Operating Systems	2-8
About the Directories for Installation and Configuration	2-9
About the Recommended Directory Structure	2-9
About the Oracle Home Directory	2-10
About the Domain Home Directory	2-11

About the Application Home Directory	2-11
Installing Multiple Products in the Same Domain	2-12
Preparing for Shared Storage	2-12
About JDK Requirements for an Oracle Fusion Middleware Installation	2-13
About Database Requirements for an Oracle Fusion Middleware Installation	2-13
Obtaining the Product Distribution	2-13
Verifying Digital Signature and Integrity of Installation Archive Files	2-14

### 3 Installing the Oracle HTTP Server Software

---

Starting the Installation Program	3-1
Navigating the Installation Screens	3-1
Verifying the Installation	3-2
Reviewing the Installation Log Files	3-2
Checking the Directory Structure	3-3
Viewing the Contents of the Oracle Home	3-3

### 4 Configuring Oracle HTTP Server in a Collocated Domain

---

Stopping the Administration Server	4-1
Starting the Configuration Wizard	4-1
Navigating the Configuration Wizard Screens to Extend the Domain	4-2
Selecting the Domain Type and Domain Home Location	4-2
Selecting the Configuration Template for Oracle HTTP Server	4-2
Specifying the Domain Mode and JDK	4-3
Specifying the Database Configuration Type	4-4
Specifying JDBC Component Schema Information	4-4
Testing the JDBC Connections	4-5
Selecting Advanced Configuration	4-5
Specifying the Oracle HTTP Server Details	4-5
Assigning System Components to Machines	4-6
Reviewing Your Configuration Specifications and Configuring the Domain	4-6
Writing Down Your Domain Home and Administration Server URL	4-6
Starting the Servers	4-7
Starting Node Manager	4-7
Starting the Administration Server	4-8
Starting the Oracle HTTP Server Instance	4-9

### 5 Configuring Oracle HTTP Server in a Standalone Domain

---

Starting the Configuration Wizard	5-1
Navigating the Configuration Wizard Screens to Configure the Domain	5-2

Selecting the Configuration Type and Domain Home Location	5-2
Selecting the Configuration Template for Oracle HTTP Server	5-2
Specifying the JDK	5-2
Adding System Components	5-3
Specifying the Oracle HTTP Server Details	5-3
Configuring Node Manager	5-4
Reviewing Your Configuration Specifications and Configuring the Domain	5-4
Starting the Instances	5-4
Starting Node Manager	5-4
Starting the Oracle HTTP Server Instance	5-5
Missing Libraries Might Cause HTTPD to Exit Without Notice	5-6

## 6 Next Steps After Configuring the Domain

---

Next Steps for Oracle HTTP Server in a WebLogic Server Domain	6-1
Next Steps for Oracle HTTP Server in a Standalone Domain	6-10
Preparing Your Environment for High Availability	6-11

## 7 Uninstalling or Reinstalling Oracle HTTP Server

---

Stopping Oracle Fusion Middleware	7-1
Uninstalling the Software	7-1
Starting the Uninstall Wizard	7-1
Navigating the Uninstall Wizard Screens	7-2
Verifying the Uninstallation of Oracle HTTP Server in a Standalone Domain	7-2
Removing the Oracle Home Directory Manually	7-2
Removing the Program Shortcuts on Windows Operating Systems	7-3
Removing the Domain and Application Data	7-3
Reinstalling the Software	7-4

## A Oracle HTTP Server Directory Structure

---

Oracle HTTP Server Directory Structure in a Collocated Domain	A-1
Oracle HTTP Server Directory Structure in a Standalone Domain	A-3

## B Updating the JDK After Installing and Configuring an Oracle Fusion Middleware Product

---

About Updating the JDK Location After Installing an Oracle Fusion Middleware Product	B-1
Updating the JDK Location in an Existing Oracle Home	B-2
Updating the JDK Location in an Existing Domain Home	B-2

# Preface

This document describes how to install and configure an Oracle HTTP Server domain.

## Audience

This document is intended for system administrators or application developers who are installing the Oracle HTTP Server. It is assumed that readers are familiar with Web technologies and have a general understanding of Windows and UNIX platforms.

## Documentation Accessibility

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

### Access to Oracle Support

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

## Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

## Related Documents

Refer to the Oracle Fusion Middleware Library for additional information.

- For Oracle HTTP Server information, see Oracle HTTP Server Documentation.
- For installation information, see Fusion Middleware Installation Documentation.
- For upgrade information, see Fusion Middleware Upgrade Documentation.
- For administration-related information, see Fusion Middleware Administration Documentation.
- For release-related information, see Fusion Middleware Release Notes.

---

## Conventions

This document uses the following text conventions:

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

---

# 1

## About the Oracle HTTP Server Installation

The standard installation for Oracle HTTP Server described in this guide creates the standard topology, which represents a sample starting topology for this product.

### About Oracle HTTP Server

Oracle HTTP Server provides a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards.

In summary, Oracle HTTP Server:

- Can be configured to act as a reverse Proxy for the Oracle WebLogic Server through the built-in Oracle WebLogic Server Proxy Plug-In
- Provides the Web Server component for Oracle Fusion Middleware
- Serves static web content, such as HTML, JavaScript, images, and dynamic web content built with CGI/FastCGI-based applications.

See Introduction to Oracle HTTP Server in *Administering Oracle HTTP Server*.

Oracle HTTP Server can be installed in either an existing Oracle home or in a standalone domain:

- In an existing Oracle home, Oracle HTTP Server is collocated with a WebLogic Server, so the Oracle HTTP Server instance can be managed like any other element of the WebLogic Server domain. Specifically, you can manage a Oracle HTTP Server instance from the Oracle Enterprise Manager Fusion Middleware Control or the WLST command line interface. See [About the Standard Installation Topology for Oracle HTTP Server in a WebLogic Server Domain](#).

You can configure a collocated Oracle HTTP Server to run in either Full-JRF or Restricted-JRF mode. See [About Full-JRF and Restricted-JRF Modes](#).

- In a standalone domain, the standalone Oracle HTTP Server topology is installed and configured without Oracle Fusion Middleware Infrastructure. See [About the Standard Installation Topology for Oracle HTTP Server in a Standalone Domain](#).

#### Note:

In a typical enterprise deployment where Oracle HTTP Server is part of the Web tier and is used for routing requests to the Application tier, Oracle recommends configuring and using Oracle HTTP Server in a standalone mode instead of collocated mode. Although collocated mode is a fully supported configuration, standalone mode provides the same routing capabilities as collocated mode and has the advantage of smaller footprint and simpler configuration. When running Oracle HTTP Server in a standalone mode, the FMW Control UI is not available for monitoring and management of Oracle HTTP Server. Users who place a high priority on Oracle HTTP Server management through FMW Control can use FMW Control in Oracle HTTP Server collocated mode configurations.



## Using the Standard Installation Topology As a Starting Point

The standard installation topology is a flexible topology that you can use as a starting point in production environments.

The information in this guide helps you to create a standard installation topology for Oracle HTTP Server. If required, you can later extend the standard installation topology to create a secure and highly available production environment, see [Next Steps After Configuring the Domain](#).

The standard installation topology represents a sample topology for this product. It is not the only topology that this product supports. See *About the Standard Installation Topology in Planning an Installation of Oracle Fusion Middleware*.

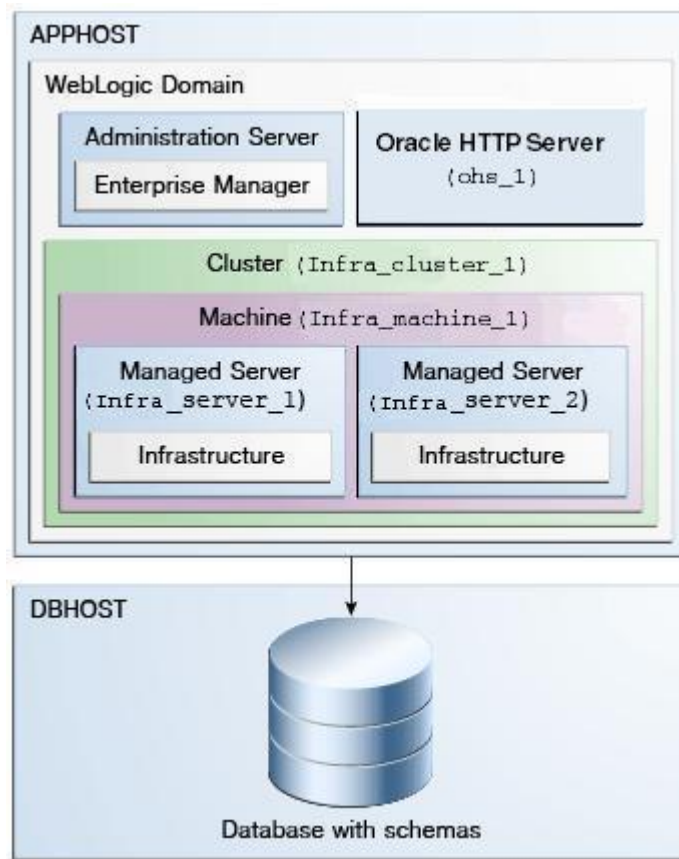
## About the Standard Installation Topology for Oracle HTTP Server in a WebLogic Server Domain

Oracle HTTP Server can be installed in an existing Oracle home, collocated with a WebLogic Server domain

In this type of installation, the Oracle HTTP Server instance can be managed like any other elements of the WebLogic Server domain. Specifically, you can manage your Oracle HTTP Server instance from the Oracle Enterprise Manager Fusion Middleware Control, the WLST Command line interface, and the Node Manager.

[Figure 1-1](#) shows the standard Oracle Fusion Middleware Infrastructure topology with an Oracle HTTP Server instance collocated in the same domain home. You can also use this topology to scale out to a high availability environment spanning multiple machines.

**Figure 1-1 Topology of an Oracle HTTP Server Installation in a WebLogic Server Domain**



For configuration instructions, see [Configuring Oracle HTTP Server in a Collocated Domain](#)

## Understanding Elements in the Standard Installation Topology Illustration

The standard installation topology typically includes common elements.

[Table 1-1](#) describes all elements of the topology illustration.

**Table 1-1 Description of Elements in the Standard Installation Topology for Oracle HTTP Server**

Element	Description and Links to Additional Documentation
APPHOST	A standard term used in Oracle documentation to refer to the machine that hosts the application tier.
DBHOST	A standard term used in Oracle documentation to refer to the machine that hosts the database.
WebLogic Domain	A logically related group of Java components (in this case, the Administration Server, Managed Servers, and other related software components). For more information, see <i>What Is an Oracle WebLogic Server Domain?</i> in <i>Understanding Oracle Fusion Middleware</i> .

**Table 1-1 (Cont.) Description of Elements in the Standard Installation Topology for Oracle HTTP Server**

Element	Description and Links to Additional Documentation
Administration Server	The central control entity of a domain which maintains the configuration objects for that domain and distributes configuration changes to the Managed Servers. For more information, see <i>What Is the Administration Server?</i> in <i>Understanding Oracle Fusion Middleware</i> .
Enterprise Manager Fusion Middleware Control	The Oracle Enterprise Manager Fusion Middleware Control is a primary tool used to manage a domain. For more information, see <i>Oracle Enterprise Manager Fusion Middleware Control</i> in <i>Understanding Oracle Fusion Middleware</i> .
Oracle HTTP Server	The Oracle HTTP Server binaries and an Oracle HTTP Server instance.
Cluster	A collection of multiple WebLogic Server instances running simultaneously and working together. For more information, see <i>Overview of Managed Servers and Managed Server Clusters</i> in <i>Understanding Oracle Fusion Middleware</i> .
Machine	A logical representation of the computer that hosts one or more WebLogic Server instances (servers). Machines are also the logical glue between the Managed Servers and the Node Manager. You must associate Managed Servers with a machine to use Node Manager to start or stop them.
Managed Server	A host for your applications, application components, web services, and their associated resources. For more information, see <i>Overview of Managed Servers and Managed Server Clusters</i> in <i>Understanding Oracle Fusion Middleware</i> .
Infrastructure	A collection of services that include: <ul style="list-style-type: none"> <li>• Metadata repository (MDS) containing metadata for Oracle Fusion Middleware components, such as the Oracle Application Developer Framework. For more information, see <i>What Is the Metadata Repository?</i> in <i>Understanding Oracle Fusion Middleware</i>.</li> <li>• Oracle Application Developer Framework (Oracle ADF)</li> <li>• Oracle Web Services Manager (OWSM)</li> </ul>

## About Full-JRF and Restricted-JRF Modes

You can configure a collocated Oracle HTTP Server to run in either Full-JRF or Restricted-JRF mode.

[Table 1-2](#) provides a comparison of these two modes.

**Table 1-2 Comparison of Full-JRF and Restricted-JRF Modes**

Operational Mode	Description	Recommended Use Case
Full Domain (or Full-JRF)	Full-JRF mode is dependent on a database connection.	This operational mode is best if your domain is accessing upper stack features, such as Oracle SOA Suite or Oracle Web Services Manager.

**Table 1-2 (Cont.) Comparison of Full-JRF and Restricted-JRF Modes**

Operational Mode	Description	Recommended Use Case
Restricted-JRF	<p>Restricted-JRF mode works without a database connection.</p> <p>Users can still manage all products via the Oracle Enterprise Manager Fusion Middleware Control, but they cannot use cross component wiring.</p>	This operational mode is best if your domain only accesses Oracle HTTP Server.

Note that migrating from Restricted-JRF to Full-JRF is not currently supported.

For a full description of what these two operational modes offer, see Domain Types in *Administering Oracle HTTP Server*.

## About the Standard Installation Topology for Oracle HTTP Server in a Standalone Domain

Oracle HTTP Server can be installed in a standalone domain in its own Oracle home.

You can create a standalone Oracle HTTP Server domain dedicated to managing system components only. A standalone configuration resides in its own Oracle home, which you create during the installation process, which means that the standalone Oracle HTTP Server topology can be installed and configured without the Oracle Fusion Middleware Infrastructure.

A standalone domain cannot be managed with Fusion Middleware Control. You can manage your standalone Oracle HTTP Server domain with the WLST command-line and other features available in a standalone domain.

[Figure 1-2](#) shows the standard Oracle HTTP Server instance in the Web tier.

**Figure 1-2 Topology of an Oracle HTTP Server in a Standalone Domain**



For configuration instructions, see [Configuring Oracle HTTP Server in a Standalone Domain](#)

[Table 1-3](#) describes the elements of this topology.

**Table 1-3 Description of the Elements in the Oracle HTTP Server Standalone Installation Topology**

Element	Description and Links to Additional Documentation
WEBHOST	This a separate host, which the sends requests to a WebLogic Server domain.

**Table 1-3 (Cont.) Description of the Elements in the Oracle HTTP Server Standalone Installation Topology**

Element	Description and Links to Additional Documentation
Standalone Domain	For more information, see Standalone Domain in <i>Administering Oracle HTTP Server</i> .
Oracle HTTP Server	The Oracle HTTP Server instance, which is created and managed by the management tools available in the standalone domain.



**Note:**

For more information on standalone domains in general, see What Is a Standalone Domain? in *Understanding Oracle Fusion Middleware*.

# 2

## Preparing to Install and Configure Oracle HTTP Server

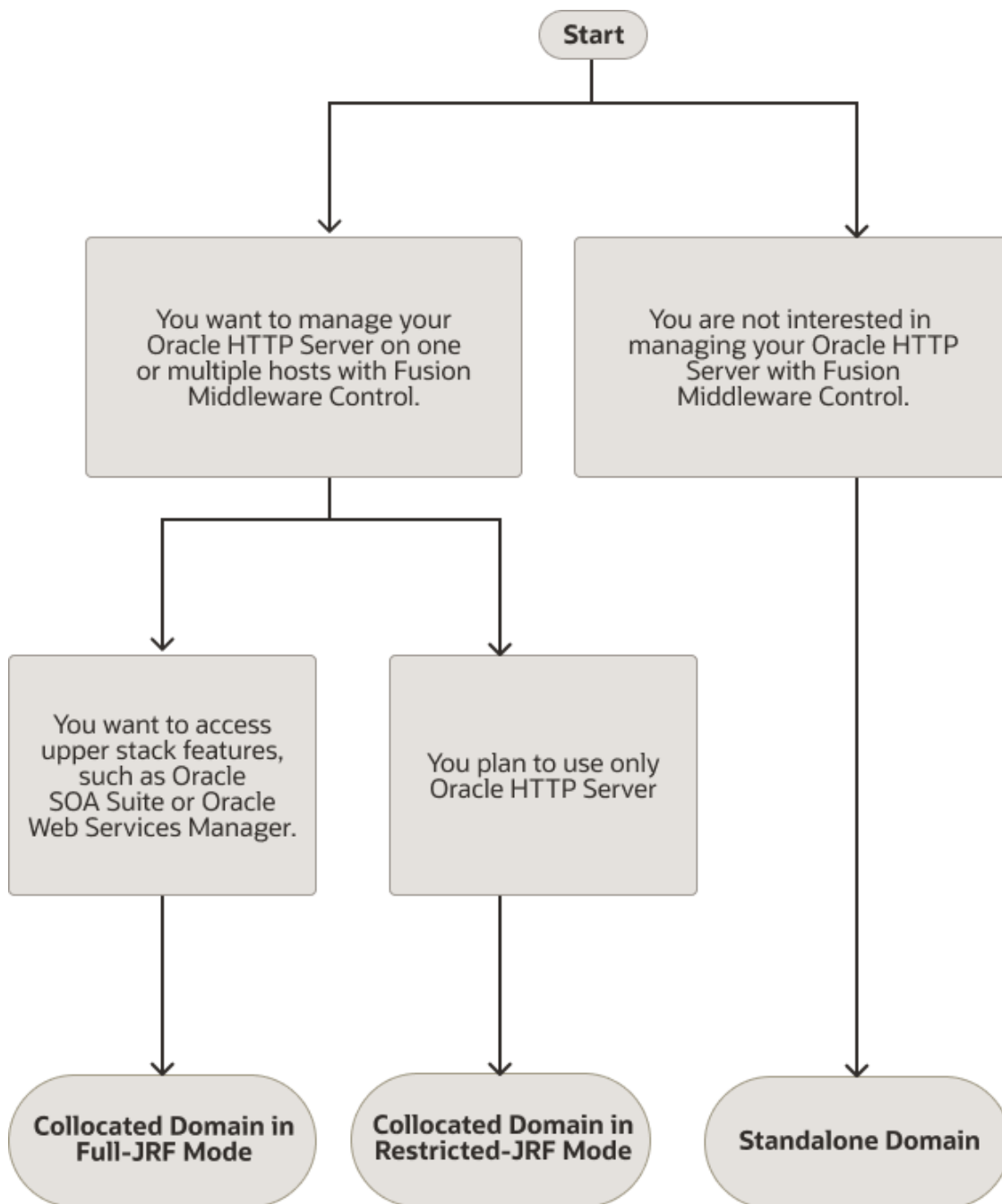
To prepare for your Oracle HTTP Server installation, verify that your system meets the basic requirements, then obtain the correct installation software.

### Selecting an Oracle HTTP Server Installation Type

You must decide whether to install Oracle HTTP Server collocated with an existing WebLogic Server or in a standalone domain.

Use the following flowchart to decide which of the two primary Oracle HTTP Server domain options to configure.

Figure 2-1 Oracle HTTP Server Domain Options



## Roadmap for Installing and Configuring Oracle HTTP Server in a Collocated Domain

This roadmap provides steps required to install and configure Oracle HTTP Server in a Collocated domain.

**Table 2-1 Roadmap for Oracle HTTP Server Installation Configured in a Collocated Domain**

Task	Description	For More Information
Install and configure the Oracle Fusion Middleware Infrastructure standard installation topology.	Before beginning the Oracle HTTP Server installation, you must install and configure Oracle Fusion Middleware Infrastructure.	See <i>Installing the Infrastructure Software in Oracle Fusion Middleware Installing and Configuring the Oracle Fusion Middleware Infrastructure</i> .
Make sure all processes associated with the Oracle Fusion Middleware Infrastructure domain are shut down.	If your Node Manager, Administration Server, Managed Servers, or any other servers are up and running, then you must shut them down before you can begin installing and configuring Oracle HTTP Server in a WebLogic Server domain.	Stopping an Oracle Fusion Middleware Environment in <i>Administering Oracle Fusion Middleware</i> . Methods for Installing Multiple Products in the Same Domain in <i>Planning an Installation of Oracle Fusion Middleware</i> .
Verify your system environment.	Before beginning the installation, verify that the minimum system and network requirements are met.	See <a href="#">Roadmap for Verifying Your System Environment</a>
Check for any mandatory patches that will be required before or after the installation.	Review the Oracle Fusion Middleware Infrastructure release notes to see if there are any mandatory patches required for the software products you are installing.	See <i>Install and Configure</i> in <i>Release Notes for Oracle Fusion Middleware Infrastructure</i> .
Obtain the appropriate distribution.	To create the topology described in this guide, obtain the Oracle HTTP Server distribution.	See <a href="#">Obtaining the Product Distribution</a> .
Determine your installation directories.	Verify that the installer can access or create the required installer directories. Also, verify that the directories exist on systems that meet the minimum requirements.	See <a href="#">Oracle HTTP Server Directory Structure</a> .
Install Oracle HTTP Server.	Run the installation program to install the software. On the Installation Type screen, select <b>Collocated HTTP Server (Managed through WebLogic server)</b> .  The installer then transfers the software to your system and updates the Oracle Fusion Middleware Infrastructure Oracle home to include the Oracle HTTP Server binaries.	See <a href="#">Installing the Oracle HTTP Server Software</a> .
Update the existing Infrastructure domain.	Use the configuration wizard to extend the Infrastructure domain to include an Oracle HTTP Server instance.  During this step, choose whether you want the Full-JRF or Restricted-JRF operation mode. See <a href="#">About Full-JRF and Restricted-JRF Modes</a> for more information about these modes.  <b>Note:</b> If you have not configured Oracle Fusion Middleware Infrastructure yet, then you can configure it at the same time as Oracle HTTP Server.	See <a href="#">Configuring Oracle HTTP Server in a Collocated Domain</a> .



**Table 2-1 (Cont.) Roadmap for Oracle HTTP Server Installation Configured in a Collocated Domain**

Task	Description	For More Information
Review post-configuration administration and configuration tasks.	Discover additional tools and resources to administer your domain and configure your domain to be highly available.	See <a href="#">Next Steps After Configuring the Domain</a> .

## Roadmap for Installing and Configuring Oracle HTTP Server in a Standalone Domain

This roadmap provides the steps required to install and configure Oracle HTTP Server in a standalone domain.

**Table 2-2 Roadmap for Oracle HTTP Server Installation Configured in a Standalone Domain**

Task	Description	For More Information
Verify your system environment.	Before beginning the installation, verify that the minimum system and network requirements are met.	See <a href="#">Roadmap for Verifying Your System Environment</a> .
Obtain the appropriate distribution.	To create the topology described in this guide, obtain the Oracle HTTP Server distribution.	See <a href="#">Obtaining the Product Distribution</a> .
Determine your installation directories.	Verify that the <i>product</i> and <i>config</i> directories are created, are accessible by the installer, and exist on systems that meet the minimum requirements.	See <a href="#">Oracle HTTP Server Directory Structure</a> .
Install Oracle HTTP Server.	Run the installation program to install the software. On the Installation Type screen, select <b>Standalone HTTP Server (Managed independently of WebLogic server)</b> .  The installer then transfers the software to your system and creates the Oracle home directory.	See <a href="#">Installing the Oracle HTTP Server Software</a> .
Create an Oracle HTTP Server Standalone Domain.	Use the configuration wizard to create and configure an Oracle HTTP Server in a standalone domain.	See <a href="#">Configuring Oracle HTTP Server in a Standalone Domain</a> .
Review post-configuration administration and configuration tasks.	Discover additional tools and resources to administer your domain and configure your domain to be highly available.	See <a href="#">Next Steps After Configuring the Domain</a> .

## Roadmap for Verifying Your System Environment

Before you begin the installation and configuration process, you must verify your system environment.

[Table 2-3](#) identifies important tasks and checks to perform to ensure that your environment is prepared to install and configure Oracle HTTP Server.

**Table 2-3 Roadmap for Verifying Your System Environment**

Task	Description	Documentation
Verify certification and system requirements.	Verify that your operating system is certified and configured for installation and configuration.	See <a href="#">Verifying Certification, System, and Interoperability Requirements</a> .
Identify a proper installation user.	Verify that the installation user has the required permissions to install and configure the software.	See <a href="#">Selecting an Installation User</a> .
Select the installation and configuration directories on your system.	Verify that you can create the necessary directories to install and configure the software, according to the recommended directory structure. Select a new, empty Oracle Home directory.	See <a href="#">About the Directories for Installation and Configuration</a> .
Install a certified JDK.	The installation program for the distribution requires a certified JDK present on your system.	See <a href="#">About JDK Requirements for an Oracle Fusion Middleware Installation</a> .
Install and configure a database for mid-tier schemas.	To configure your WebLogic domain, you must have access to a certified database that is configured for the schemas required by Oracle HTTP Server.	See <a href="#">About Database Requirements for an Oracle Fusion Middleware Installation</a> .

## Verifying Certification, System, and Interoperability Requirements

Oracle recommends that you use the certification matrix and system requirements documents with each other to verify that your environment meets the requirements for installation.

### 1. Verifying that your environment meets certification requirements:

Ensure that you install your product on a supported hardware and software configuration.

Oracle has tested and verified the performance of your product on all certified systems and environments. Whenever new certifications are released, they are added to the certification document right away. New certifications can be released at any time. Therefore, the certification documents are kept outside the documentation libraries and are available on Oracle Technology Network.

### 2. Using the system requirements document to verify certification:

Oracle recommends that you use the *Oracle Fusion Middleware System Requirements and Specifications* document to verify that the certification requirements are met. System requirements can change in the future. Therefore, the system requirement documents are kept outside of the documentation libraries and are available on Oracle Technology Network.

### 3. Verifying interoperability among multiple products:

To learn how to install and run multiple Fusion Middleware products from the same release or mixed releases with each other, see Oracle Fusion Middleware Interoperability and Compatibility in *Understanding Interoperability and Compatibility*.

## Selecting an Installation User

The user who installs and configures your system must have the required permissions and privileges.

### About User Permissions

The user who installs a Fusion Middleware product owns the files and has certain permissions on the files.

- Read and write permissions on all non-executable files (for example, `.jar`, `.properties`, or `.xml`). All other users in the same group as the file owner have read permissions only.
- Read, write, and execute permissions on all executable files (for example, `.exe`, `.sh`, or `.cmd`). All other users in the same group as the file owner have read and execute permissions only.

This means that someone other than the person who installs the software can use the installed binaries in the Oracle home directory to configure a domain or set of Fusion Middleware products.

During configuration, the files generated by the configuration process are owned by the user who ran the Configuration Wizard. This user has the same permissions as described above for the installation user. However, security-sensitive files are not created with group permissions. Only the user that created the domain has read and write permissions and can administer the domain.

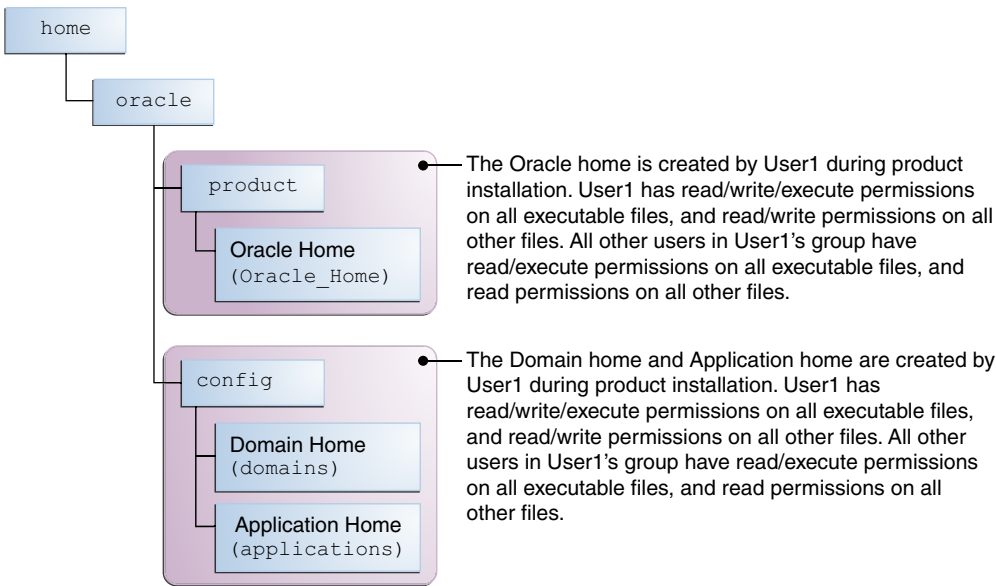
Consider the following examples:

- **Example 1: A Single User Installs the Software and Configures the Domain**

[Figure 2-2](#) explains the file permissions where the same user installs the software and configures the domain.

To ensure proper permissions and privileges for all files, Oracle recommends that the same owner perform both tasks: install the Oracle Fusion Middleware product and configure the WebLogic Server domain by using the Configuration Wizard.

**Figure 2-2 Directory Structure When you manage a product installation - a Single User Installs the Software and Configures the Domain**

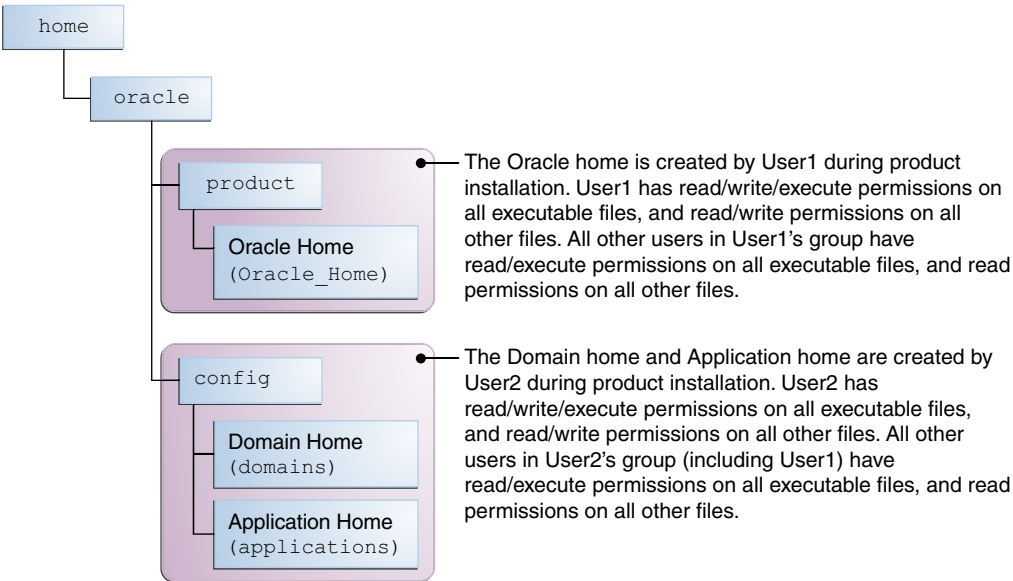


If the user who creates the domain is different than the user who installed the software, then both users must have the same privileges, as shown in the next example.

- **Example 2: The Oracle Home Directory and Domain are Created by Different Users**

Figure 2-3 explains the file permissions where one user creates the Oracle home and another user configures the domain.

**Figure 2-3 Directory Structure when Different Users Install the Software and Configure the Domain**





**Note:**

Certain domain files do not have group permissions. For example, `cwallet.sso`.

Consider the following points before you run the installer:

- On UNIX operating systems, Oracle recommends that you set `umask` to `027` on your system before you install the software. This ensures that the file permissions are set properly during installation. Use the following command:

```
umask 027
```

You must enter this command in the same terminal window from which you plan to run the product installer.

- On UNIX operating systems, do not run the installation program as a `root` user. If you run the installer as a root user, the startup validation may fail and you cannot continue the installation.
- When you manage a product installation (for example, applying patches), use the same user ID that you used to install the product.  
When you manage a domain (for example, starting managed Servers), use the same user ID that you used to create the domain.
- On Windows operating systems, you must have administrative privileges to install the product. See [Verifying the Installation User has Administrator Privileges on Windows Operating Systems](#).

## About Non-Default User Permissions on Linux or UNIX Operating Systems

Changing the default permission setting reduces the security of the installation and your system. Oracle does not recommend that you change the default permission settings.

If other users require access to a particular file or executable, use the Linux or UNIX `sudo` command or other similar commands to change the file permissions.

Refer to your Linux or UNIX operating system Administrator's Guide or contact your operating system vendor, if you need further assistance.

## Verifying That the Installation User Has Administrator Privileges on Windows Operating Systems

To update the Windows Registry, you must have administrator privileges.

By default, users with the administrator privilege sign in to the system with regular privileges, but can request elevated permissions to perform administrative tasks.

To perform a task with elevated privileges:

1. Find the Command Prompt icon, either from the Start menu or the Windows icon in the lower-left corner.
2. Right-click **Command Prompt** and select **Run as administrator**.

This opens a new command prompt window, and all actions performed in this window are done with administrator privileges.

 **Note:**

If you have User Access Control enabled on your system, you may see an additional window asking you to confirm this action. Confirm and continue with this procedure.

3. Perform the desired task.

For example, to start the product installer:

For a jar file, enter:

```
java -jar distribution_name.jar
```

For an executable (.exe, .bin, or .sh file), enter:

```
distribution_name.exe
```

## About the Directories for Installation and Configuration

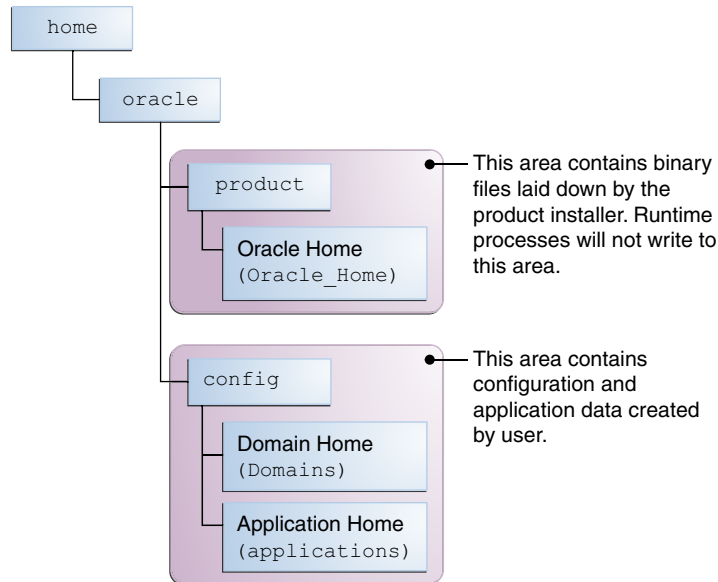
During the installation and domain configuration process, you must plan on providing the locations for these directories: Oracle home, Domain home, and the Application home.

## About the Recommended Directory Structure

Oracle recommends specific locations for the Oracle Home, Domain Home, and Application Home.

Oracle recommends a directory structure similar to the one shown in [Figure 2-4](#).

**Figure 2-4 Recommended Oracle Fusion Middleware Directory Structure**



A base location (Oracle base) should be established on your system (for example, /home/oracle). From this base location, create two separate branches, namely, the `product` directory and the `config` directory. The `product` directory should contain the product binary files and all

the Oracle home directories. The `config` directory should contain your domain and application data.

Oracle recommends that you do not keep your configuration data in the Oracle home directory; if you upgrade your product to another major release, you are required to create an Oracle home for binary files. You must also make sure that your configuration data exists in a location where the binary files in the Oracle home have access.

The `/home/oracle/product` (for the Oracle home) and `/home/oracle/config` (for the application and configuration data) directories are used in the examples throughout the documentation; be sure to replace these directories with the actual directories on your system.

## About the Oracle Home Directory

When you install any Oracle Fusion Middleware product, you must use an Oracle home directory.

This directory is a repository for common files that are used by multiple Fusion Middleware products installed on the same machine. These files ensure that Fusion Middleware operates correctly on your system. They facilitate checking of cross-product dependencies during installation. For this reason, you can consider the Oracle home directory a *central support directory* for all Oracle Fusion Middleware products installed on your system.

Fusion Middleware documentation refers to the Oracle home directory as `ORACLE_HOME`.

### Oracle Home Considerations

Keep the following in mind when you create the Oracle home directory and install the Oracle Fusion Middleware products:

- Do not include spaces in the name of your Oracle home directory; the installer displays an error message if your Oracle home directory path contains spaces.
- You can install only one instance of each Oracle Fusion Middleware product in a single Oracle home directory. If you need to maintain separate versions of a product on the same machine, each version must be in its own Oracle home directory.

Although you can have several different products in a single Oracle home, only one version of each product can be in the Oracle home.

### Multiple Home Directories

Although in most situations, a single Oracle home directory is sufficient, it is possible to create more than one Oracle home directory. For example, you need to maintain multiple Oracle home directories in the following situations:

- You prefer to maintain separate development and production environments, with a separate product stack for each. With two directories, you can update your development environment without modifying the production environment until you are ready to do so.
- You want to maintain two different versions of a Fusion Middleware product at the same time. For example, you want to install a new version of a product while keeping your existing version intact. In this case, you must install each product version in its own Oracle home directory.
- You need to install multiple products that are not compatible with each other. See Oracle Fusion Middleware Interoperability and Compatibility in *Understanding Interoperability and Compatibility*.

 **Note:**

If you create more than one Oracle home directory, you must provide non-overlapping port ranges during the configuration phase for each product.

## About the Domain Home Directory

The Domain home is the directory where domains that you configure are created.

The default Domain home location is `ORACLE_HOME/user_projects/domains/domain_name`.

 **Note:**

Oracle strongly recommends that you do not use the default location. Put your Domain home *outside* of the Oracle home directory, for example, in `/home/oracle/config/domains`.

The `config` directory should contain domain and application data. Oracle recommends a separate domain directory so that new installs, patches, and other operations update the `ORACLE_HOME` only, *not* the domain configuration.

See [About the Recommended Directory Structure](#) for more on the recommended directory structure and locating your Domain home.

Fusion Middleware documentation refers to the Domain home directory as `DOMAIN_HOME` and includes all folders up to and including the domain name. For example, if you name your domain `exampledomain` and locate your domain data in the `/home/oracle/config/domains` directory, the documentation would use `DOMAIN_HOME` to refer to `/home/oracle/config/domains/exampledomain`.

## About the Application Home Directory

The Application home is the directory where applications for domains you configure are created.

The default Application home location is `ORACLE_HOME/user_projects/applications/domain_name`. However, Oracle strongly recommends that you locate your Application home *outside* of the Oracle home directory; if you upgrade your product to another major release, you must create an Oracle home for binary files.

See [About the Recommended Directory Structure](#) for more on the recommended directory structure and locating your Application home.

Fusion Middleware documentation refers to the Application home directory as `APPLICATION_HOME` and includes all folders up to and including the domain name. For example, if you name your domain `exampledomain` and you locate your application data in the `/home/oracle/config/applications` directory, the documentation uses `APPLICATION_HOME` to refer to `/home/oracle/config/applications/exampledomain`.



## Installing Multiple Products in the Same Domain

There are two methods to install and configure multiple products in one domain. This is also known as *extending* a domain.

- **Method 1.**

Install and configure Product A, including creating the schemas and starting all servers in the domain to verify a successful domain configuration.

This is the method used in all installation guides in the Fusion Middleware library. You can repeat this process for as many products as necessary. It allows you to validate one product at a time and add more products incrementally.

To install Product B in the same domain as Product A:

1. Stop all servers to prevent any updates to the domain while you add the new product.  
See Starting and Stopping Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.
2. Follow the instructions in the installation guide for Product B, including creating the necessary schemas.
3. Run the Configuration Wizard to configure the domain.

During configuration, the Configuration Wizard automatically detects the components that have been installed and offers you the option to extend the existing Product A domain to include Product B.

- **Method 2.**

Install all of the required products, then create the schemas for all of the products. After you create the schemas, configure the domain by using the necessary product templates, then start all the servers.

This method of creating a multi-product domain may be slightly faster than Method 1; however, the installation guides in the Fusion Middleware library do not provide specific instructions for this method of domain creation.

 **See Also:**

- To update WebLogic domains, see Updating WebLogic Domains in *Creating WebLogic Domains Using the Configuration Wizard*.
- For important information regarding the ability of Oracle Fusion Middleware products to function with previous versions of other Oracle Fusion Middleware, Oracle, or third-party products, see Oracle Fusion Middleware Interoperability and Compatibility in *Understanding Interoperability and Compatibility*.

## Preparing for Shared Storage

Oracle Fusion Middleware allows you to configure multiple WebLogic Server domains from a single Oracle home. This allows you to install the Oracle home in a single location on a shared volume and reuse the Oracle home for multiple host installations.

If you plan to use shared storage in your environment, see Using Shared Storage in *High Availability Guide* for more information.

## About JDK Requirements for an Oracle Fusion Middleware Installation

Most Fusion Middleware products are in `.jar` file format. These distributions do not include a JDK. To run a `.jar` distribution installer, you must have a certified JDK installed on your system.

Make sure that the JDK is installed *outside* of the Oracle home. If you install the JDK under the Oracle home, you may encounter problems when you try to perform tasks in the future. Oracle recommends that you locate your JDK installation in the `/home/oracle/products/jdk` directory.

Platform-specific distributions have a `.bin` (for Linux operating systems) or `.exe` (for Windows operating systems) installer; in these cases, a platform-specific JDK is in the distribution and you do not need to install a JDK separately. However, you may need to upgrade this JDK to a more recent version, depending on the JDK versions that are certified.

Always verify the required JDK version by reviewing the certification information on the *Oracle Fusion Middleware Supported System Configurations* page for Oracle Fusion Middleware 14c (14.1.2.0.0).

To download the required JDK, navigate to the following URL and download the Java SE JDK:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

## About Database Requirements for an Oracle Fusion Middleware Installation

Many Oracle Fusion Middleware products require database schemas prior to configuration. If you do not already have a database where you can install these schemas, you must install and configure a certified database.

To find a certified database for your operating system, see the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page on *Technical Resources from Oracle*.

To make sure that your database is properly configured for schema creation, see Repository Creation Utility Requirements in the *Oracle Fusion Middleware System Requirements and Specifications* document.

After your database is properly configured, you use the Repository Creation Utility (RCU) to create product schemas in your database. This tool is available in the Oracle home for your Oracle Fusion Middleware product. See About the Repository Creation Utility in *Creating Schemas with the Repository Creation Utility*.

## Obtaining the Product Distribution

The Oracle HTTP Server distribution is available as a standalone executable installation program that includes a Java Runtime Environment.

For more information about the the Oracle HTTP Server distribution, see Obtaining Product Distributions in *Planning an Installation of Oracle Fusion Middleware*.

# Verifying Digital Signature and Integrity of Installation Archive Files

Oracle digitally signs the installation archive files with Oracle certificates to ensure the integrity of the packages before you deploy them in your environments.

Use the Java utility `jarsigner` to verify the integrity of your installation archive files. You can verify the integrity of the installation archive files before you extract the installation files.

## Quick Verification

To quickly verify the installation archive files, use the `jarsigner` command with the `-verify` option:

1. Go to the directory where you have downloaded the installation archive files.
2. Run this command to check your installation archive file:

```
jarsigner -verify installation_archive_file
```

For example, to check the Oracle Fusion Middleware Infrastructure archive:

```
jarsigner -verify fmw_14.1.2.0.0_infrastructure.jar
```

```
jar verified.
```

## Detailed Certificate Information

If you want detailed certificate information, then use the `-verbose:summary` and `-certs` along with the `-verify` option.

1. Go to the directory where you have downloaded the installation archive files.
2. Run this command to check your installation archive file:

```
jarsigner -verify -verbose:summary -certs installation_archive_file
```

For example, to check the Oracle Fusion Middleware Infrastructure image:

```
jarsigner -verify -verbose:summary -certs fmw_14.1.2.0.0_infrastructure.jar
```

The output is similar to the following:

```
2237119 Fri Dec 6 07:02:30 UTC 2023 META-INF/MANIFEST.MF

>>> Signer
  X.509, CN="Oracle America, Inc.", O="Oracle America, Inc.",
L=Redwood City, ST=California, C=US
  [
    Signature algorithm: SHA256withRSA, 3072-bit key
    [certificate is valid from 12/19/24 12:00 AM to 12/19/25 11:59 PM]
    X.509, CN=DigiCert Trusted G4 Code Signing RSA4096 SHA384 2021 CA1,
O="DigiCert, Inc.", C=US
```

```
[
Signature algorithm: SHA384withRSA, 4096-bit key
[certificate is valid from 4/29/24 12:00 AM to 4/28/36 11:59 PM]
X.509, CN=DigiCert Trusted Root G4, O=DigiCert Inc, C=US
[
Signature algorithm: SHA384withRSA, 4096-bit key
[trusted certificate]
>>> TSA
X.509, CN=DigiCert Timestamp 2024 - 2, O=DigiCert, C=US
[
Signature algorithm: SHA256withRSA, 4096-bit key
[certificate is valid from 9/21/24 12:00 AM to 11/21/33 11:59 PM]
X.509, CN=DigiCert Trusted G4 RSA4096 SHA256 TimeStamping CA,
O="DigiCert, Inc.", C=US
[
Signature algorithm: SHA256withRSA, 4096-bit key
[certificate is valid from 3/23/24 12:00 AM to 3/22/37 11:59 PM]
X.509, CN=DigiCert Trusted Root G4, O=DigiCert Inc, C=US
[
Signature algorithm: SHA384withRSA, 4096-bit key
[certificate is valid from 8/1/24 12:00 AM to 11/9/31 11:59 PM]

2237281 Fri Feb 17 07:02:32 UTC 2024 META-INF/ORACLE_C.SF (and 1
more)

(Signature related entries)

0 Fri Feb 17 05:41:24 UTC 2023 OPatch/ (and 1897 more)

(Directory entries)

2977 Tue Dec 20 08:02:16 UTC 2024 OPatch/README.txt (and 20199 more)

[entry was signed on 2/17/24 7:02 AM]
>>> Signer
X.509, CN="Oracle America, Inc.", O="Oracle America, Inc.",
L=Redwood City, ST=California, C=US
[
Signature algorithm: SHA256withRSA, 3072-bit key
[certificate is valid from 8/19/24 12:00 AM to 8/19/25 11:59 PM]
X.509, CN=DigiCert Trusted G4 Code Signing RSA4096 SHA384 2021 CA1,
O="DigiCert, Inc.", C=US
[
Signature algorithm: SHA384withRSA, 4096-bit key
[certificate is valid from 4/29/24 12:00 AM to 4/28/36 11:59 PM]
X.509, CN=DigiCert Trusted Root G4, O=DigiCert Inc, C=US
[
Signature algorithm: SHA384withRSA, 4096-bit key
[trusted certificate]
>>> TSA
X.509, CN=DigiCert Timestamp 2024 - 2, O=DigiCert, C=US
[
Signature algorithm: SHA256withRSA, 4096-bit key
[certificate is valid from 9/21/24 12:00 AM to 11/21/33 11:59 PM]
X.509, CN=DigiCert Trusted G4 RSA4096 SHA256 TimeStamping CA,
O="DigiCert, Inc.", C=US
```

```
[  
Signature algorithm: SHA256withRSA, 4096-bit key  
[certificate is valid from 3/23/24 12:00 AM to 3/22/37 11:59 PM]  
X.509, CN=DigiCert Trusted Root G4, O=DigiCert Inc, C=US  
[  
Signature algorithm: SHA384withRSA, 4096-bit key  
[certificate is valid from 8/1/24 12:00 AM to 11/9/31 11:59 PM]
```

```
s = signature was verified  
m = entry is listed in manifest  
k = at least one certificate was found in keystore  
i = at least one certificate was found in identity scope
```

```
- Signed by "CN="Oracle America, Inc.", O="Oracle America, Inc.",  
L=Redwood City, ST=California, C=US"  
Digest algorithm: SHA-256  
Signature algorithm: SHA256withRSA, 3072-bit key  
Timestamped by "CN=DigiCert Timestamp 2024 - 2, O=DigiCert, C=US" on Fri  
Feb 17 07:02:33 UTC 2024  
Timestamp digest algorithm: SHA-256  
Timestamp signature algorithm: SHA256withRSA, 4096-bit key
```

```
jar verified.
```

```
The signer certificate will expire on 2025-12-19.  
The timestamp will expire on 2031-11-09.
```

# 3

## Installing the Oracle HTTP Server Software

Follow the steps in this section to install the Oracle HTTP Server software. Before beginning the installation, ensure that you have verified the prerequisites and completed all steps covered in [Preparing to Install and Configure Oracle HTTP Server](#).

### Starting the Installation Program

You can start the installation program on UNIX or Windows.

To start the installation program:

1. Sign in to the host system.
2. Go to the directory where you have extracted the contents of product distribution archive file.
3. Enter the following command:
  - (UNIX) `./fmw_14.1.2.0.0_ohs_linux64.bin`
  - (Windows) `setup_fmw_14.1.2.0.0_ohs_win64.exe`



#### Note:

You will not be able to execute `./fmw_14.1.2.0.0_ohs_linux64.bin` if it does not have execute permission. Make sure to check and grant execute permission before running this command.

When the installation program appears, you are ready to begin the installation.

### Navigating the Installation Screens

The installer shows a series of screens where you verify or enter information.

[Table 3-1](#) lists the order in which installer screens appear. If you need additional help with an installation screen, click **Help**.

**Table 3-1** Install Screens

Screen	Description
Installation Inventory Setup	<p>On Linux or Unix operating systems, this screen opens if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location.</p> <p>See About the Oracle Central Inventory in <i>Installing Software with the Oracle Universal Installer</i>.</p> <p>This screen does not appear on Windows operating systems.</p>

**Table 3-1 (Cont.) Install Screens**

Screen	Description
Welcome	Review the information to make sure that you have met all the prerequisites, then click <b>Next</b> .
Auto Updates	Select to skip automatic updates, select patches, or search for the latest software updates, including important security updates, through your My Oracle Support account.
Installation Location	Specify your Oracle home directory location. You can click <b>View</b> to verify and ensure that you are installing in the correct Oracle home.
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. To view the list of tasks that gets verified, select <b>View Successful Tasks</b> . To view log details, select <b>View Log</b> . If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click <b>Rerun</b> to try again. To ignore the error or the warning message and continue with the installation, click <b>Skip</b> (not recommended).
Installation Summary	Use this screen to verify installation options you selected. If you want to save these options to a response file, click <b>Save Response File</b> and enter the response file location and name. The response file collects and stores all the information that you have entered, and enables you to perform a silent installation (from the command line) at a later time. Click <b>Install</b> to begin the installation.
Installation Progress	This screen shows the installation progress. When the progress bar reaches 100% complete, click <b>Finish</b> to dismiss the installer, or click <b>Next</b> to see a summary.
Installation Complete	This screen displays the Installation Location and the Feature Sets that are installed. Review this information and click <b>Finish</b> to close the installer.

## Verifying the Installation

After you complete the installation, verify whether it was successful by completing a series of tasks.

## Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that the installer did not encounter any problems.

By default, the installer writes logs files to the `Oracle_Inventory_Location/logs` directory on Linux operating systems.

In case of Windows operating systems, the installer writes logs files to the `Oracle_Inventory_Location\logs` directory.

For a description of the log files and where to find them, see Installation Log Files in *Installing Software with the Oracle Universal Installer*.

## Checking the Directory Structure

The contents of your installation vary based on the options you selected during the installation.

To verify that your installation completed successfully, you can compare your directory structure to the one shown in [Oracle HTTP Server Directory Structure](#).

For more information about the directory structure after installation, see [What are the Key Oracle Fusion Middleware Directories?](#) in *Understanding Oracle Fusion Middleware*.

## Viewing the Contents of the Oracle Home

You can view the contents of the Oracle home directory by using the `viewInventory` script.

See [Viewing the Contents of an Oracle Home](#) in *Installing Software with the Oracle Universal Installer*.



# 4

## Configuring Oracle HTTP Server in a Collocated Domain

After you have installed Oracle HTTP Server, you can extend an existing Infrastructure domain with Oracle HTTP Server.

### Note:

- Perform the configuration steps in this section only if you selected **Collocated HTTP Server (Managed through WebLogic server)** in the Installation Type screen. If you selected **Standalone HTTP Server (Managed independently of WebLogic server)**, see [Configuring Oracle HTTP Server in a Standalone Domain](#).
- For information about Oracle HTTP Server domain configuration options, see [About Oracle HTTP Server](#).

If you have not created the Oracle Fusion Middleware Infrastructure domain, you can create it at the same time as Oracle HTTP Server. For instructions, see *Configuring the Oracle Fusion Middleware Infrastructure Domain* in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*. However, in **Selecting Advanced Configurations**, ensure that you select **System Components** too.

## Stopping the Administration Server

Before starting the configuration wizard, it is important to shut down the Administration Server. This prevents any configuration locks, saves, or activations from occurring during the configuration of the domain.

To stop the Administration Server, go to the `DOMAIN_HOME/bin` directory.

- On UNIX operating systems:  

```
./stopWebLogic.sh
```
- On Windows operating systems:  

```
stopWebLogic.cmd
```

You may be prompted for the login credentials of the Administrator user you provided when you configured the Oracle Fusion Middleware Infrastructure domain.

## Starting the Configuration Wizard

Start the Configuration Wizard to begin configuring a domain.

To start the Configuration Wizard:

1. Change to the following directory:  
(UNIX) `ORACLE_HOME/oracle_common/common/bin`  
(Windows) `ORACLE_HOME\oracle_common\common\bin`  
where `ORACLE_HOME` is your 14c (14.1.2.0.0) Oracle home.
2. Enter the following command:  
(UNIX) `./config.sh`  
(Windows) `config.cmd`

## Navigating the Configuration Wizard Screens to Extend the Domain

Use the Configuration Wizard to extend an existing Infrastructure domain with an Oracle HTTP Server instance.

For more information, see [Understand How Domain Mode Affects the Default Security Configuration](#).



### Note:

When you are adding an Oracle HTTP Server instance to an existing domain, verify that the ports you are assigning to any new components are available. The Configuration Wizard does not perform automatic port allocation.

The following sections step through the Configuration Wizard:

### Selecting the Domain Type and Domain Home Location

You must specify the domain to be updated with Oracle HTTP Server.

To specify the Domain type and Domain home directory:

1. On the Configuration Type screen, select **Update an existing domain**.
2. In the Domain Location field, select the Domain home location specified while configuring the Infrastructure domain.

For more information about this screen, see [Configuration Type](#) in *Creating WebLogic Domains Using the Configuration Wizard*.

### Selecting the Configuration Template for Oracle HTTP Server

Use the Templates screen to select the template you require.

On the Templates screen, select one of the following templates:

- For a Full-JRF configuration, select `Oracle HTTP Server (Collocated) - 14.1.2.0.0 [ohs]`.
- For a Restricted-JRF configuration, select `Oracle HTTP Server (Restricted JRF) - 14.1.2.0.0 [ohs]`.

Selecting this template automatically selects the following as dependencies:

- Oracle Enterprise Manager-Restricted JRF
- Oracle Restricted JRF
- WebLogic Coherence Cluster Extension

If you are unsure what operational mode you want, see [About Full-JRF and Restricted-JRF Modes](#) for an overview of both modes.

For more information about this screen, see Templates in *Creating WebLogic Domains Using the Configuration Wizard*

## Specifying the Domain Mode and JDK

Use the Domain Mode and JDK screen to specify the domain mode and Java Development Kit (JDK) for your production environment.

On the Domain Mode and JDK screen:

- Select **Production** in the **Domain Mode** field.

### Note:

As of WebLogic Server 14.1.2.0.0, when you select **Production** mode, WebLogic Server automatically sets some of the security configurations of **Secured Production** to more secure values. However, there are certain security configurations (such as SSL/TLS) that require manual configuration. See *Using Secured Production Mode in Administering Security for Oracle WebLogic Server*.

The port configuration in Oracle HTTP Server should be consistent with the port configuration in WebLogic Server. For example, if the SSL ports are enabled and Listen ports (non-SSL) are disabled in Oracle HTTP Server, then the SSL ports should also be enabled in WebLogic Server.

If you want to disable the more secure default settings, then you may select **Disable Secure Mode**. This will enable the non-SSL listen ports.

If you want to retain the more secure default settings of **Secured Production** mode in general, but want to change which ports (listen ports, SSL listen ports, or administration ports) will be enabled by default in your domain, then you may:

- Leave **Disable Secure Mode** unselected, and
- Change the default port selections under **Enable or Disable Default Ports for Your Domain**

For more information, see *Understand How Domain Mode Affects the Default Security Configuration in Securing a Production Environment for Oracle WebLogic Server*.

- Select the **Oracle HotSpot JDK** in the **JDK** field.

For more information about this screen, see Domain Mode and JDK in *Creating WebLogic Domains Using the Configuration Wizard*.

## Specifying the Database Configuration Type

Use the Database Configuration type screen to specify details about the database and database schema.

On the Database Configuration type screen, all fields are pre-populated, because you already configured the domain to reference the Fusion Middleware schemas that are required for the Infrastructure domain. Verify and ensure that credentials in all the fields are the same that you have provided while configuring Oracle Fusion Middleware Infrastructure.

Verify the database connection information, then select **Get RCU Configuration**. The following output in the Connection Result Log indicates that the operation succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK
```

Successfully Done.

For more information about the schema installed when RCU is run, see Understanding the Service Table Schema in *Creating Schemas with the Repository Creation Utility*.

For more information about this screen, see Database Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*.

### Note:

The following JDBC-related tasks are required for Full-JRF mode only; for Restricted-JRF mode, you can skip them.

## Specifying JDBC Component Schema Information

Use the JDBC Component Schema screen to verify or specify details about the database schemas.

Verify that the values populated on the JDBC Component Schema screen are correct for all schemas. If you selected **RCU Data** on the previous screen, the schema table should already be populated appropriately.

For an Autonomous Transaction Processing database (both Autonomous Transaction Processing-Dedicated (ATP-D) and Autonomous Transaction Processing Shared (ATP-S)), specify the connection credentials using only the **Connection URL String** option, and enter the connect string in the following format:

```
@TNS_alias?TNS_ADMIN=<path of the wallet files, ojdbc.properties, and
tnsnames.ora>
```

In the connect string, you must pass `TNS_alias` as the database service name found in `tnsnames.ora`, and `TNS_ADMIN` property to the location of the wallet files, `ojdbc.properties`, and `tnsnames.ora`.

Example connect string for Autonomous Transaction Processing-Dedicated (ATP-D) database:

```
@dbname_tp?TNS_ADMIN=/users/test/wallet_dbname/
```

Example connect string for Autonomous Transaction Processing Shared (ATP-S) database:

```
@dbname_tp?TNS_ADMIN=/users/test/wallet_dbname/
```

For high availability environments, see the following sections in *High Availability Guide* for additional information on configuring data sources for Oracle RAC databases:

- Configuring Active GridLink Data Sources with Oracle RAC
- Configuring Multi Data Sources

See JDBC Component Schema in *Creating WebLogic Domains Using the Configuration Wizard* for more details about this screen.

## Testing the JDBC Connections

Use the JDBC Component Schema Test screen to test the data source connections.

A green check mark in the Status column indicates a successful test. If you encounter any issues, see the error message in the Connection Result Log section of the screen, fix the problem, then try to test the connection again.

By default, the schema password for each schema component is the password you specified while creating your schemas.

For more information about this screen, see JDBC Component Schema Test in *Creating WebLogic Domains Using the Configuration Wizard*.

## Selecting Advanced Configuration

Use the Advanced Configuration screen to complete the domain configuration.

On the Advanced Configuration screen, select:

- System Components

## Specifying the Oracle HTTP Server Details

Use the OHS Server screen to configure the Oracle HTTP Server instance in the domain.

In the Listen Address field, specify the IP address of the host on which the Oracle HTTP Server instance will reside. Do not use `localhost`. See *Specifying the Listen Address* in *Creating WebLogic Domains Using the Configuration Wizard*.

All of the remaining fields are pre-populated, but you can change the values as required. For example, verify that the default port numbers are not used by other services.

 **Note:**

If you change the admin host from default IP address 127.0.0.1 to any other IP address, you must ensure the following:

- Set `ServerName` directive in `admin.conf` to the host name or domain name that maps to the IP address that you have chosen.
- Create an SSL certificate with a common name matching the host name or domain name set up using the `ServerName` directive.

For more information, see *Configuring SSL for Admin Port* in the *OHS Admin Guide*.

For more information about this screen, see OHS Server in *Creating WebLogic Domains Using the Configuration Wizard*.

## Assigning System Components to Machines

Use the Assign System Components to Machines screen to assign Oracle HTTP Server system components to the machine that you defined when you configured the initial Fusion Middleware Infrastructure domain.

On the Assign System Components to Machines screen:

1. In the **Machine** list box, select the machine (for example, `infra_machine_1`) to which you want to assign the Oracle HTTP Server instance.
2. Assign the Oracle HTTP Server instance to the selected machine in one of the following ways:
  - Double-click `ohs1` in the **System Components** list box.
  - Select the `ohs1` in the **System Components** list box and click the right arrow.

You can associate an Oracle HTTP Server machine to an existing machine or create a new machine. You typically create additional machines when you are planning to scale out the topology to multiple host computers. For more information, see *Scaling Out a Topology (Machine Scale Out)* in *High Availability Guide*.

## Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen has detailed configuration information for the domain you are about to extend.

Review each item on the screen and verify that the information is correct. To make any changes, go back to a screen by clicking the **Back** button or selecting the screen in the navigation pane. Domain update does not start until you click **Update**.

For more information about this screen, see *Configuration Summary* in *Creating WebLogic Domains Using the Configuration Wizard*.

## Writing Down Your Domain Home and Administration Server URL

The End of Configuration screen shows information about the domain you just configured.

Make a note of the following items because you need them later:

- Domain Location
- Administration Server URL

You need the domain location to access scripts that start Node Manager and Administration Server, and you need the URL to access the Administration Server.

Click **Finish** to dismiss the Configuration Wizard.

## Starting the Servers

After configuration is complete, start Node Manager, then the WebLogic Administration Server and Managed Servers.

### Note:

Depending on your existing security settings, you may need to perform additional configuration before you can start and manage a domain with secured production mode enabled. Specifically, you will need to add additional parameters when starting the Administration and Managed Servers. For more information, see *Using Secured Production Mode Administering Security for Oracle WebLogic Server*.

For more information on additional tools you can use to manage your domain, see *Overview of Oracle Fusion Middleware Administration Tools* in *Administering Oracle Fusion Middleware*.

## Starting Node Manager

To start the per-domain Node Manager:

1.
  - (UNIX) Go to the `DOMAIN_HOME/bin` directory.
  - (Windows) Go to the `DOMAIN_HOME\bin` directory.
2. Enter the following command:
  - (UNIX) Using `nohup` and `nm.out` as an example output file:

```
nohup ./startNodeManager.sh > LOG_DIR/nm.out&
```

where `LOG_DIR` is the location of directory in which you want to store the log files.

- (Windows) `startNodeManager.cmd`

### Note:

On Windows operating systems, Oracle recommends that you configure Node Manager to run as a startup service. This allows Node Manager to start up automatically each time the system is restarted.

See *Running Node Manager as a Startup Service* in *Administering Node Manager for Oracle WebLogic Server*.

## Starting the Administration Server

The procedures in this section describe how to start the Administration Server using the WLST command line or a script. You can also use the Oracle Fusion Middleware Control and the Oracle WebLogic Server Remote Console. See Starting and Stopping Administration and Managed Servers and Node Manager in *Administering Oracle Fusion Middleware*.

To start the Administration Server:

 **Note:**

When using secured production mode, you must provide additional parameters to start the Administration Server. See Connecting to the Administration Server using WLST in *Administering Security for Oracle WebLogic Server*.

1. **(Optional)** When using **Production Mode**, you can create a *boot.properties* file before starting the Administration Server and provide necessary permissions. This file can be created to bypass the need to provide a username and password when starting the Administration Server. For more information, see Creating a Boot Identity File for an Administration Server in *Administering Server Startup and Shutdown for Oracle WebLogic Server*.

2. Go to the `DOMAIN_HOME/bin` directory.

3. Enter the following command:

- (UNIX)  
`./startWebLogic.sh`
- (Windows)  
`startWebLogic.cmd`

If you selected **Production Mode** on the Domain Mode and JDK screen when you created the domain, and you did not create the optional *boot.properties* file, you see a prompt for the Administrator user login credentials as provided on the Administrator Account screen.

4. Open a browser and verify that the Administration Server is up and running. The default port values will vary depending on how you configured your domain. The Enable SSL Listen Port is enabled by default, but the default values may change. For a list of default values, see Port Numbers by Product and Component.

`https://<Host_Name>:<port>`

5. Verify that all servers in the domain have unique port values. From the WebLogic Remote Console, you can review the **Local Administration Port Override** fields for each managed server and verify that each has a unique value. If one or more ports is using the same value, then you must change them before starting the managed servers. For more information about changing port values, see Connect to an Administration Server in the Oracle WebLogic Remote Console.



 **Note:**

The WebLogic Server Administration Console has been removed. For comparable functionality, you should use the WebLogic Remote Console. For more information, see Oracle WebLogic Remote Console.

## Starting the Oracle HTTP Server Instance

To start the Oracle HTTP Server instance from Fusion Middleware Control:

1. Sign in to Fusion Middleware Control:

```
http://administration_server_host:administration_server_port/em
```

The Administration Server host and port number were provided in [Writing Down Your Domain Home and Administration Server URL](#).

The login credentials were provided on the Administrator Account screen while installing.

2. Click **HTTP Server** in the navigation panel on the left side.
3. Select the Oracle HTTP Server instance you want to start (for example, ohs1) and click **Start Up** from the top panel.

To start the Oracle HTTP Server from the command line:

- **On UNIX**

1. Go to the `DOMAIN_HOME/bin` directory.
2. Modify `JAVA_OPTIONS`, if the domain is in the secure mode. See [Understand How Domain Mode Affects the Default Security Configuration](#).

 **Note:**

You must perform this step only if the domain is in secure production mode.

3. Run the following command:

```
./startComponent.sh system_component_name
```

where `system_component_name` is the Oracle HTTP Server system component name you provided while configuring the domain.

For example:

```
./startComponent.sh ohs1
```

- **On Windows**

1. Go to the `DOMAIN_HOME\bin` directory.
2. Run the following command:

```
startComponent.cmd system_component_name
```

where `system_component_name` is the Oracle HTTP Server system component name you provided while configuring the domain.

For example:

```
startComponent.cmd ohs1
```

For more information about managing Oracle HTTP Server using Oracle Enterprise Manager Fusion Middleware Control, see Manage Oracle HTTP Server in *Administering Oracle Fusion Middleware with Fusion Middleware Control*.

For other tools and instructions, see Starting Oracle HTTP Server Instances in *Administering Oracle HTTP Server*.

# 5

## Configuring Oracle HTTP Server in a Standalone Domain

After you have installed Oracle HTTP Server, you can configure it in a standalone domain. For more information on other methods available for domain creation, see *Additional Tools for Creating, Extending, and Managing WebLogic Domains in Creating WebLogic Domains Using the Configuration Wizard*.



### Note:

Perform the configuration steps in this section only if you have selected **Standalone HTTP Server (Managed independently of WebLogic server)** in the Installation Type screen. If you have selected **Collocated HTTP Server (Managed through WebLogic server)**, see [Configuring Oracle HTTP Server in a Collocated Domain](#).

For information about Oracle HTTP Server domain configuration options, see [About Oracle HTTP Server](#).

The following sections step through the Configuration Wizard:

### Starting the Configuration Wizard

Start the Configuration Wizard to begin configuring a domain.

To start the Configuration Wizard:

1. Change to the following directory:  
(UNIX) `ORACLE_HOME/oracle_common/common/bin`  
(Windows) `ORACLE_HOME\oracle_common\common\bin`  
where `ORACLE_HOME` is your 14c (14.1.2.0.0) Oracle home.
2. Enter the following command:  
(UNIX) `./config.sh`  
(Windows) `config.cmd`

# Navigating the Configuration Wizard Screens to Configure the Domain

Use the Configuration Wizard to configure Oracle HTTP Server in a standalone domain.



## Note:

When you are creating an Oracle HTTP Server instance for a new domain, verify that the ports you are assigning to your components are available. The Configuration Wizard does not perform automatic port allocation.

## Selecting the Configuration Type and Domain Home Location

Use the Configuration Type screen to select a Domain home directory location, optimally outside the Oracle home directory.

Oracle recommends that you locate your Domain home in accordance with the directory structure in *What Are the Key Oracle Fusion Middleware Directories?* in *Understanding Oracle Fusion Middleware*, where the Domain home is located outside the Oracle home directory. This directory structure helps avoid issues when you need to upgrade or reinstall software.

To specify the Domain type and Domain home directory:

1. On the Configuration Type screen, select **Create a new domain**.
2. In the Domain Location field, specify your Domain home directory.

For more details about this screen, see Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*.

## Selecting the Configuration Template for Oracle HTTP Server

Use the Templates screen to select the template you require.

On the Templates screen, select the following template:

- Oracle HTTP Server (Standalone) - 14.1.2.0.0 [ohs]

For more information about this screen, see Templates in *Creating WebLogic Domains Using the Configuration Wizard*

## Specifying the JDK

Use the JDK Selection screen to select the JDK for the domain.

Select the **Oracle HotSpot JDK** in the **JDK** field.

If you select **Other JDK Location**, select only a JDK that is supported on the platform you are using. Always verify the required JDK version by reviewing the certification information on the *Oracle Fusion Middleware Supported System Configurations* page for Oracle Fusion Middleware 14c (14.1.2.0.0).

## Adding System Components

Use the System Components screen to specify details about the new Oracle HTTP Server instance.

On the System Components screen:

1. Specify `ohs1` in the **System Component** field.
2. Specify `OHS` in the **Component Type** field.
3. **Restart Interval Seconds:** Specify the number of seconds to wait before attempting a restart if an application is not responding.
4. **Restart Delay Seconds:** Specify the number of seconds to wait between restart attempts.

### Note:

You can rename the Oracle HTTP Server system component name (for example, `ohs_1`), when you rerun the configuration wizard and select **Update an existing domain** in the Configuration Type screen. But when you do this, the original Oracle HTTP Server component name and all customized values, like port designation, gets deleted and a new system component with default values get created. So, avoid renaming the Oracle HTTP Server component, unless you want the default settings.

For more information about this screen, see System Components in *Creating WebLogic Domains Using the Configuration Wizard*.

## Specifying the Oracle HTTP Server Details

Use the OHS Server screen to configure the Oracle HTTP Server instance in the domain.

In the Listen Address field, specify the IP address of the host on which the Oracle HTTP Server instance will reside. Do not use `localhost`. See Specifying the Listen Address in *Creating WebLogic Domains Using the Configuration Wizard*.

All of the remaining fields are pre-populated, but you can change the values as required. For example, verify that the default port numbers are not used by other services.

### Note:

If you change the admin host from default IP address `127.0.0.1` to any other IP address, you must ensure the following:

- Set `ServerName` directive in `admin.conf` to the host name or domain name that maps to the IP address that you have chosen.
- Create an SSL certificate with a common name matching the host name or domain name set up using the `ServerName` directive.

For more information, see Configuring SSL for Admin Port in the *OHS Admin Guide*.

For more information about this screen, see OHS Server in *Creating WebLogic Domains Using the Configuration Wizard*.

## Configuring Node Manager

Use the Node Manager screen to select the type of Node Manager you want to configure, along with the Node Manager credentials.

Select **Per Domain Default Location** as the Node Manager type, then specify Node Manager credentials.

For more information about this screen, see Node Manager in *Creating WebLogic Domains Using the Configuration Wizard*.

For more information about Node Manager types, see About Node Manager in *Administering Node Manager for Oracle WebLogic Server*.

## Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen shows detailed configuration information for the domain you are about to create.

Review each item on the screen and verify that the information is correct. To make any changes, go back to a screen by clicking the **Back** button or selecting the screen in the navigation pane. Domain creation does not start until you click **Create**.

For more details about options on this screen, see Configuration Summary in *Creating WebLogic Domains Using the Configuration Wizard*.

## Starting the Instances

After configuration is complete, you can use tools to manage your domain.

- For more information on additional tools you can use to manage your domain, see Overview of Oracle Fusion Middleware Administration Tools in *Administering Oracle Fusion Middleware*.
- If you are attempting to create an Oracle HTTP Server instance that uses a TCP port in the reserved range (typically less than 1024), then you must perform some extra configuration to allow the server to bind to privileged ports. For more information, see Starting Oracle HTTP Server Instances on a Privileged Port (Unix Only) in *Administering Oracle HTTP Server*.

## Starting Node Manager

To start the per-domain Node Manager:

1.
  - (UNIX) Go to the `DOMAIN_HOME/bin` directory.
  - (Windows) Go to the `DOMAIN_HOME\bin` directory.
2. Enter the following command:
  - (UNIX) Using `nohup` and `nm.out` as an example output file:

```
nohup ./startNodeManager.sh > LOG_DIR/nm.out&
```

where `LOG_DIR` is the location of directory in which you want to store the log files.

- (Windows) `startNodeManager.cmd`

 **Note:**

On Windows operating systems, Oracle recommends that you configure Node Manager to run as a startup service. This allows Node Manager to start up automatically each time the system is restarted.

See Running Node Manager as a Startup Service in *Administering Node Manager for Oracle WebLogic Server*.

## Starting the Oracle HTTP Server Instance

To start the Oracle HTTP Server instance:

- **On UNIX**
  1. Go to the `DOMAIN_HOME/bin` directory.
  2. Modify `JAVA_OPTIONS`, if the domain is in the secure mode. See *Understand How Domain Mode Affects the Default Security Configuration*.

 **Note:**

You must perform this step only if the domain is in secure production mode.

3. Run the following command:

```
./startComponent.sh system_component_name
```

where `system_component_name` is the Oracle HTTP Server system component name you provided while configuring the domain.

For example:

```
./startComponent.sh ohs1
```

- **On Windows**
  1. Go to the `DOMAIN_HOME\bin` directory.

2. Run the following command:

```
startComponent.cmd system_component_name
```

where `system_component_name` is the Oracle HTTP Server system component name you provided while configuring the domain.

For example:

```
startComponent.cmd ohs1
```

See Starting Oracle HTTP Server Instances from the Command Line in *Administering Oracle HTTP Server*.

## Missing Libraries Might Cause HTTPD to Exit Without Notice

On the Windows platform, Oracle HTTP Server requires certain Microsoft software to be installed in order to function properly.

If the required software is not installed, Oracle HTTP Server may exit with no output or generate an error message, depending on registry settings.

Make sure that you have installed all packages recommended by Microsoft. You can download packages from <http://www.microsoft.com>.



# 6

## Next Steps After Configuring the Domain

After you configure a product domain, there are additional tasks that you may want to perform.

### Next Steps for Oracle HTTP Server in a WebLogic Server Domain

After configuring Oracle HTTP Server in a WebLogic Server domain, there are additional tasks that you may want to perform.




**Note:**

A WebLogic Server domain can be scaled out to multiple physical machines and be centrally managed by the administration server. See Oracle HTTP Server Release Topologies in *Administering Oracle HTTP Server*.

**Table 6-1 Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain**

Task	Description	More Information
Getting familiar with Fusion Middleware administration tools	Get familiar with the various tools available, which you can use to manage your environment.	<p>See:</p> <ul style="list-style-type: none"> <li><a href="#">Oracle HTTP Server Directory Structure in a Collocated Domain</a></li> <li><a href="#">Oracle HTTP Server Directory Structure in a Standalone Domain</a></li> </ul> <p>Overview of Oracle Fusion Middleware Administration Tools in <i>Administering Oracle Fusion Middleware</i>.</p>



**N**  
**o**  
**t**  
**e**  
**:**  
**T**  
**h**  
**e**  
**W**  
**e**  
**b**  
**L**  
**o**  
**g**  
**i**  
**c**  
**S**  
**e**  
**r**  
**v**  
**e**  
**r**  
**A**  
**d**  
**m**  
**i**  
**n**  
**i**  
**s**  
**t**  
**r**  
**a**  
**t**  
**i**  
**o**  
**n**  
**C**  
**o**  
**n**  
**s**  
**o**  
**l**  
**e**

Table 6-1 (Cont.) Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain

Task	Description	More Information
		has been removed. For comparable functionality, you will use this

**Table 6-1 (Cont.) Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain**

Task	Description	More Information
		e W e b L o g i c R e m o t e C o n s o l e .

**Table 6-1 (Cont.) Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain**

Task	Description	More Information
Starting and stopping products and servers	Learn how to start and stop Oracle Fusion Middleware, including the Administration Server, Managed Servers, and components.	See Starting and Stopping Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .


 **N**  
**o**  
**t**  
**e**  
**:**  
 D  
 e  
 p  
 e  
 n  
 d  
 i  
 n  
 g  
 o  
 n  
 y  
 o  
 u  
 r  
 e  
 x  
 i  
 s  
 t  
 i  
 n  
 g  
 s  
 e  
 c  
 u  
 r  
 i  
 t  
 y  
 s  
 e  
 t  
 t

Table 6-1 (Cont.) Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain

Task	Description	More Information
		ings, you may need to perform additional configuration before you

**Table 6-1 (Cont.) Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain**

Task	Description	More Information
		o u c a n m a n a g e a d o m a i n w i t h s e c u r e p r o d u c t i o n m o d e n a b l e d . F o r

Table 6-1 (Cont.) Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain

Task	Description	More Information
		more information, see <a href="#">Connecting to the Administration Server</a>



**Table 6-1 (Cont.) Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain**

Task	Description	More Information
Configuring Secure Sockets Layer (SSL)	Learn how to set up secure communications among between Oracle Fusion Middleware components using SSL.	See Configuring SSL in Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Deploying Applications	Learn how to deploy your applications to Oracle Fusion Middleware.	See Deploying Applications in <i>Administering Oracle Fusion Middleware</i> .
Monitoring Oracle Fusion Middleware	Learn how to keep track of the status of Oracle Fusion Middleware components.	See Monitoring Oracle Fusion Middleware in <i>Administering Oracle Fusion Middleware</i> .
Understanding Backup and Recovery Procedures	Learn the recommended backup and recovery procedures for Oracle Fusion Middleware.	See Introducing Backup and Recovery in <i>Administering Oracle Fusion Middleware</i> .

Using WebLogic Console

**Table 6-1 (Cont.) Basic Administration Tasks for Oracle HTTP Server in a WebLogic Server Domain**

Task	Description	More Information
Tuning and configuring Coherence for your topology.	The standard installation topology includes a Coherence cluster that contains storage-enabled Managed Coherence Servers. This configuration is a good starting point for using Coherence, but depending upon your specific requirements, consider tuning and reconfiguring Coherence to improve performance in a production environment.	See <i>Configuring and Managing Coherence Clusters in Administering Clusters for Oracle WebLogic Server</i> . For information on tuning Coherence, see <i>Performance Tuning in Administering Oracle Coherence</i> . For information on storing HTTP session data in Coherence, see <i>Using Coherence Web with WebLogic Server in Administering HTTP Session Management with Oracle Coherence*Web</i> . For more about creating and deploying Coherence applications, see <i>Getting Started in Developing Oracle Coherence Applications for Oracle WebLogic Server</i> .
Wiring Oracle HTTP Server to the Administration Server.	Learn how to connect Oracle HTTP Server to the Administration Server so that you can access Fusion Middleware Control Console through the Oracle HTTP Server.	See <i>Wiring Oracle HTTP Server to the Administration Server in Administering Oracle Fusion Middleware</i> .
Routing Applications Through Oracle HTTP Server to Oracle WebLogic Server.	Perform these steps to connect Oracle HTTP Server so that requests are routed through Oracle HTTP Server to Oracle WebLogic Server	See <i>Routing Applications Through Oracle HTTP Server to Oracle WebLogic Server in Administering Oracle Fusion Middleware</i> .
Routing requests to managed servers in a WebLogic Server domain	Learn how to configure Oracle WebLogic Server Proxy Plug-In to proxy requests from Oracle HTTP Server to Oracle WebLogic Server.	See <i>Configuring the Plug-In for Oracle HTTP Server</i> .

## Next Steps for Oracle HTTP Server in a Standalone Domain

After configuring Oracle HTTP Server in a standalone domain, there are additional tasks that you may want to perform.

**Table 6-2 Basic Administration Tasks for Oracle HTTP Server in a Standalone Domain**

Task	Description	More Information
Getting familiar with Oracle HTTP Server management tools	Get familiar with the various tools available, which you can use to manage your environment.	See Understanding Oracle HTTP Server Management Tools in <i>Administering Oracle HTTP Server</i> .
Starting and stopping Oracle HTTP Server instances	Learn how to start and stop Oracle HTTP Server instances in a standalone domain.	See Performing Basic Oracle HTTP Server Tasks in <i>Administering Oracle HTTP Server</i> .
Routing requests to managed servers in a WebLogic Server domain	Learn how to configure Oracle WebLogic Server Proxy Plug-In to proxy requests from Oracle HTTP Server to Oracle WebLogic Server.	See Configuring the Oracle WebLogic Server Proxy Plug-In Manually.

## Preparing Your Environment for High Availability

Scaling out for high availability requires additional steps.

[Table 6-3](#) provides a list of tasks to perform if you want to scale out your standard installation environment for high availability.

### Note:

BAM domains that were created using WLST, and will be used in a high availability configuration, require additional provisioning scripts after the installation. The default / internal Data Objects are missing in BAM Composer when the domain is created using WLST and the scripts provide the pre-seeded data that is required for high availability BAM domains. For more information, [My Oracle Support document ID 2190789.1](#).

**Table 6-3 Tasks Required to Prepare Your Environment for High Availability**

Task	Description	More Information
Scaling out to multiple host computers	To enable high availability, it is important to provide failover capabilities to another host computer. That way, if one computer goes down, your environment can continue to serve the consumers of your deployed applications.	See Scaling Out a Topology (Machine Scale Out) in <i>High Availability Guide</i> .
Configuring high availability for your Web Tier components.	If you have added a Web tier front-end, then you must configure the Web Tier for high availability, as well as the WebLogic Server software.	See Configuring High Availability for Web Tier Components in <i>HTTP Server Administration Guide</i> .
Setting up a front-end load balancer	You can use a load balancer to distribute requests across servers more evenly.	See Server Load Balancing in a High Availability Environment in <i>High Availability Guide</i> .

**Table 6-3 (Cont.) Tasks Required to Prepare Your Environment for High Availability**

<b>Task</b>	<b>Description</b>	<b>More Information</b>
Configuring Node Manager	Node Manager enables you to start, shut down, and restart the Administration Server and Managed Server instances from a remote location. This document assumes you have configured a per-domain Node Manager. Review the Node Manager documentation, for information on advanced Node Manager configuration options and features.	See Advanced Node Manager Configuration in <i>Administering Node Manager for Oracle WebLogic Server</i> .

# 7

## Uninstalling or Reinstalling Oracle HTTP Server

Follow the instructions in this section to uninstall or reinstall Oracle HTTP Server.

### Note:

This task applies only to *standalone* Oracle HTTP Server. You cannot uninstall Oracle HTTP Server when it is collocated in a WebLogic Server domain, as partial deconfiguration of a configured domain is not supported.

Oracle recommends that you always use the instructions in this section to remove the software. If you try to remove the software manually, you may encounter problems when you try to reinstall the software again at a later time. Following the procedures in this section ensures that the software is properly removed.

## Stopping Oracle Fusion Middleware

Before running the Uninstall Wizard, Oracle recommends that you stop all servers and processes associated with the Oracle home you are going to remove.

See *Stopping an Oracle Fusion Middleware Environment* in *Administering Oracle Fusion Middleware*.

## Uninstalling the Software

Follow the instructions in this section to start the Uninstall Wizard and remove the software.

### Note:

This task applies only to *standalone* Oracle HTTP Server. You cannot uninstall Oracle HTTP Server when it is collocated in a WebLogic Server domain, as partial deconfiguration of a configured domain is not supported.

If you want to uninstall the product in a silent (command-line) mode, see *Running the Oracle Universal Installer for Silent Uninstallation* in *Installing Software with the Oracle Universal Installer*.

## Starting the Uninstall Wizard

To start the Uninstall Wizard:

1. Change to the following directory:  
(UNIX) `ORACLE_HOME/oui/bin`

(Windows) `ORACLE_HOME\oui\bin`

2. Enter the following command:

(UNIX) `./deinstall.sh`

(Windows) `deinstall.cmd`

## Navigating the Uninstall Wizard Screens

The Uninstall Wizard shows a series of screens to confirm the removal of the software.

[Table 7-1](#) describes the screens in the Uninstall Wizard. For information, click **Help** on the screen.

**Table 7-1 Uninstall Wizard Screens and Descriptions**

Screen	Description
Welcome	Introduces you to the product Uninstall Wizard.
Uninstall Summary	Shows the Oracle home directory and its contents that are uninstalled. Verify that this is the correct directory.  If you want to save these options to a response file, click <b>Save Response File</b> and enter the response file location and name. You can use the response file later to uninstall the product in silent (command-line) mode. See <i>Running the Oracle Universal Installer for Silent Uninstall in Installing Software with the Oracle Universal Installer</i> .  Click <b>Deinstall</b> , to begin removing the software.
Uninstall Progress	Shows the uninstallation progress.
Uninstall Complete	Appears when the uninstallation is complete. Review the information on this screen, then click <b>Finish</b> to close the Uninstall Wizard.

## Verifying the Uninstallation of Oracle HTTP Server in a Standalone Domain

When Oracle HTTP Server is installed in a standalone domain, you may need to manually delete directories and files.

To verify the uninstallation of Oracle HTTP Server in a standalone domain:

1. Go to the `ORACLE_HOME` directory.
2. Enter `ls` to list the directories and files in `ORACLE_HOME`.
3. If any files or directories exist in `ORACLE_HOME`, delete them by entering the following command:

```
rm -rf
```

## Removing the Oracle Home Directory Manually

After you uninstall the software, you must manually remove your Oracle home directory and any existing subdirectories that the Uninstall Wizard did not remove.

For example, if your Oracle home directory is `/home/Oracle/product/ORACLE_HOME` on Linux operating systems, enter the following commands:

```
cd /home/Oracle/product
rm -rf ORACLE_HOME
```

On Windows operating systems, if your Oracle home directory is `C:\Oracle\Product\ORACLE_HOME`, use a file manager window and navigate to the `C:\Oracle\Product` directory. Right-click on the `ORACLE_HOME` folder and select **Delete**.

## Removing the Program Shortcuts on Windows Operating Systems

On Windows operating systems, you must also manually remove the program shortcuts; the Deinstallation Wizard does not remove them for you.

To remove the program shortcuts on Windows:

1. Change to the following directory: `C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Oracle\ORACLE_HOME\Product`
2. If you only have one product installed in your Oracle home, delete the `ORACLE_HOME` directory. If you have multiple products installed in your Oracle home, delete all products before you delete the `ORACLE_HOME` directory.

## Removing the Domain and Application Data

After you uninstall the software, you must remove the domain and application data.

To remove the domain and application data:

1. Manually remove your Domain home directory. For example:

On Linux operating systems, if your Domain home directory is `/home/Oracle/config/domains/ohs_domain`, enter the following command:

```
cd /home/Oracle/config/domains
rm -rf ohs_domain
```

On Windows operating systems, if your Domain home directory is `C:\Oracle\Config\domains\ohs_domain`, use a file manager window and navigate to the `C:\Oracle\Config\domains` directory. Right-click on the `ohs_domain` folder and select **Delete**.

2. Manually remove your Application home directory. For example:

On Linux operating systems, if your Application home directory is `/home/Oracle/config/applications/ohs_domain`, enter the following commands:

```
cd /home/Oracle/config/applications
rm -rf ohs_domain
```

On Windows operating systems, if your Application home directory is `C:\Oracle\Config\applications\ohs_domain`, use a file manager window and navigate to the `C:\Oracle\Config\applications` directory. Right-click on the `ohs_domain` folder and select **Delete**.

3. Back up the `domain-registry.xml` file in your Oracle home, then edit the file and remove the line associated with the domain that you are removing. For example, to remove the `ohs_domain`, find the following line and remove it:

```
<domain location="/home/Oracle/config/domains/ohs_domain"/>
```

Save and exit the file when you are finished.

## Reinstalling the Software

You can reinstall your software into the same Oracle home as a previous installation only if you uninstalled the software by following the instructions in this section, including manually removing the Oracle home directory.

When you reinstall, you can then specify the same Oracle home as your previous installation.

Consider the following cases where the Oracle home is not empty:

- Installing in an existing Oracle home that contains the same feature sets.

The installer warns you that the Oracle home that you specified during installation already contains the same software you are trying to install.

- Installing in an existing, non-empty Oracle home.

For example, suppose you chose to create your Domain home or Application home somewhere inside your existing Oracle home. This data is not removed when you uninstall a product, so if you try to reinstall into the same Oracle home, the installer does not allow it. Your options are:

- Uninstall your software from the Oracle home (as this section describes) and then remove the Oracle home directory. After you uninstall the software and remove the Oracle home directory, you can reinstall and reuse the same Oracle home location. Any domain or application data that was in the Oracle home must be re-created.
- Select a different Oracle home directory.



# A

## Oracle HTTP Server Directory Structure

A directory structure is created on your system at the end of your Oracle HTTP Server installation and domain configuration.

### Oracle HTTP Server Directory Structure in a Collocated Domain

[Figure A-1](#) shows an example of the basic directory structure after installation and domain configuration of Oracle HTTP Server in a Collocated domain is complete. Note that figure shows only important directories. There are other directories and files created too.

This example shows the recommended directory structure, as described in Understanding the Recommended Directory Structure in *Planning an Installation of Oracle Fusion Middleware*. Your directory structure may differ slightly based on where you choose to create these directories on your system.

Figure A-1 Sample Oracle HTTP Server Directory Structure in a Collocated Domain

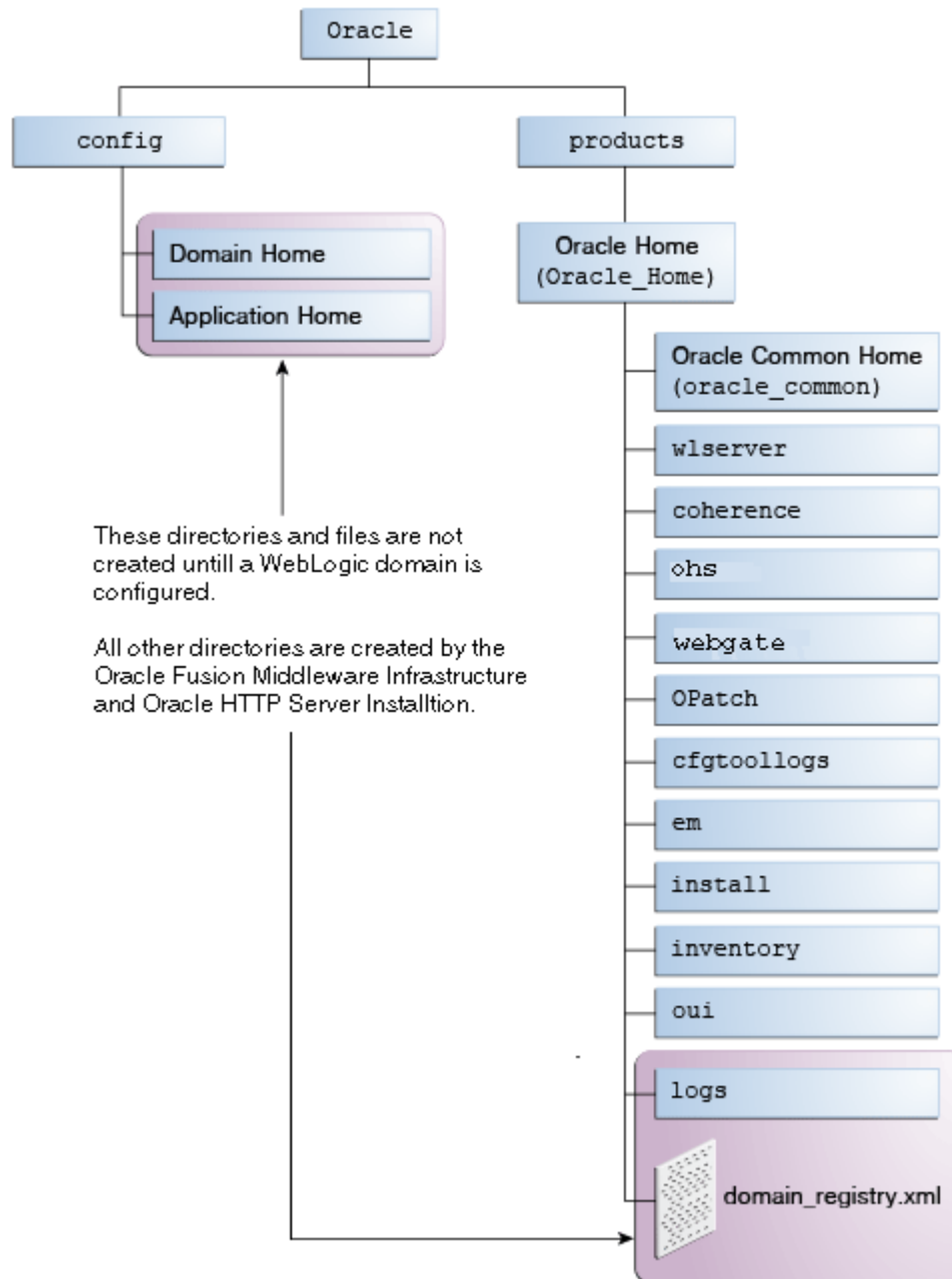


Table A-1 describes these directories in more details.

Table A-1 Directory Description for Oracle HTTP Server in a WebLogic Server Domain

Directory or file	Description
cfgtoollogs	This directory contains the log files for your installation and configuration transactions.
coherence	This directory contains the Coherence product files.
install	This directory contains install-related files and scripts.

**Table A-1 (Cont.) Directory Description for Oracle HTTP Server in a WebLogic Server Domain**

Directory or file	Description
inventory	This directory contains information about the components, feature sets, and patches installed in this Oracle home directory.
ohs	This directory contains the Oracle HTTP Server binaries.
OPatch	This directory contains OPatch and supported files. OPatch is a tool used to patch Oracle Fusion Middleware software.
oracle_common	This directory contains the binary and library files required for Oracle Enterprise Manager Fusion Middleware Control, Java Required Files (JRF), and Oracle HTTP Server.
orainst.loc	Text file containing the location of your Oracle inventory, and also the group ID of the user creating the inventory.
oui	This directory contains files used by the Oracle Universal Installer, including the uninstallation program. If you need to run the Oracle Universal Installer again after the product is installed, you should do so from this directory.
webgate	This directory contains the Oracle HTTP Server Webgate binaries for Oracle Access Manager (OAM).
wlserver	This directory contains a small portion of webLogic Server binaries that are used for management functionality. This directory is also referred to as the WebLogic home directory, or <i>WL_HOME</i> .
domain-registry.xml	This registry file contains the location of all domains currently registered with this WebLogic Server installation. Whenever you add a new domain, it is registered in this file.

## Oracle HTTP Server Directory Structure in a Standalone Domain

Figure A-2 shows an example of the basic directory structure after installation and domain configuration of Oracle HTTP Server in a standalone domain is complete. Note that figure shows only important directories. There are other directories and files created too.

This example shows the recommended directory structure, as described in Understanding the Recommended Directory Structure in *Planning an Installation of Oracle Fusion Middleware*. Your directory structure may differ slightly based on where you choose to create these directories on your system.

Figure A-2 Sample Oracle HTTP Server Directory Structure in a Standalone Domain

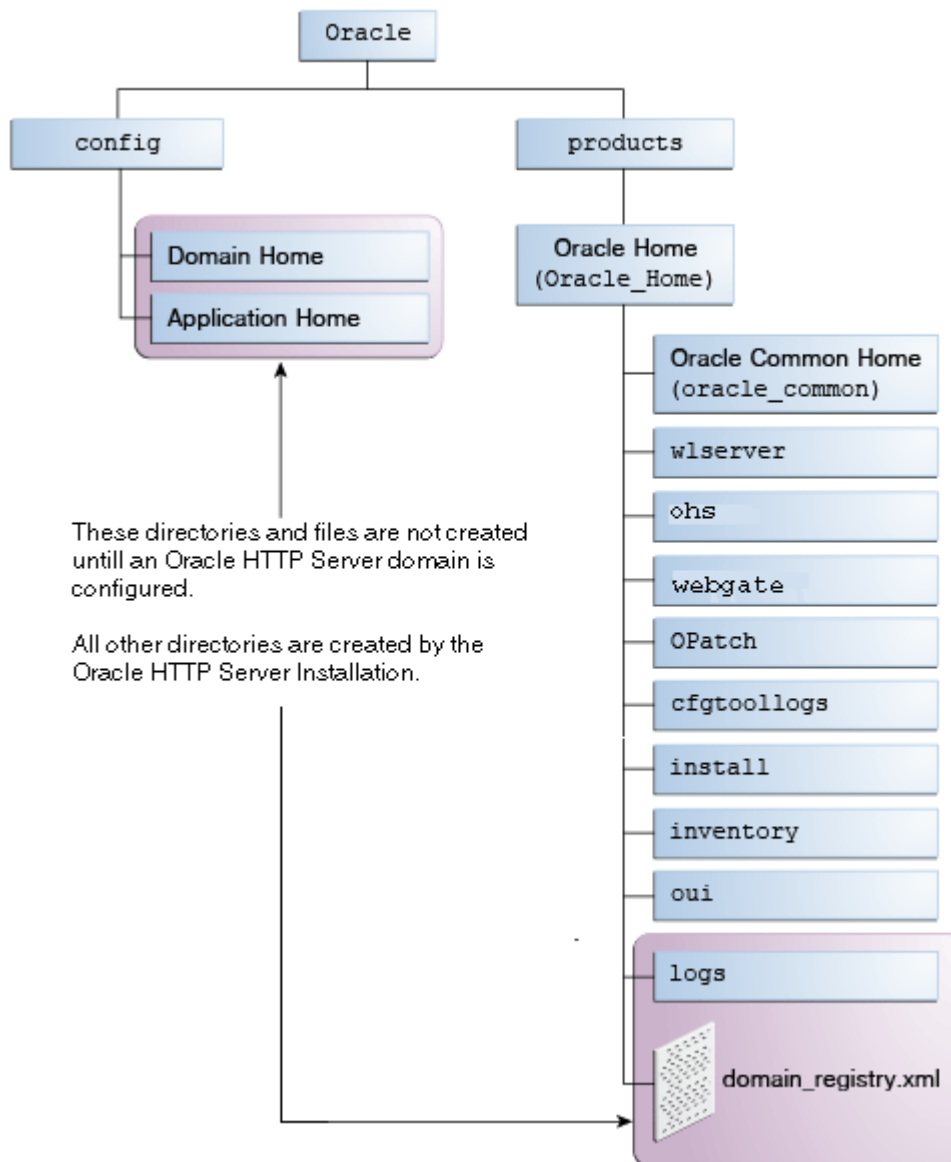


Table A-2 describes these directories in more details.

Table A-2 Directory Description for Oracle HTTP Server in a Standalone Domain

Directory or file	Description
cfgtoollogs	This directory contains the log files for your installation and configuration transactions.
install	This directory contains install-related files and scripts.
inventory	This directory contains information about the components, feature sets, and patches installed in this Oracle home directory.
ohs	This directory contains the Oracle HTTP Server binaries.
OPatch	This directory contains OPatch and supported files. OPatch is a tool used to patch Oracle Fusion Middleware software.

**Table A-2 (Cont.) Directory Description for Oracle HTTP Server in a Standalone Domain**

Directory or file	Description
oracle_common	This directory contains the binary and library files required for Oracle HTTP Server and WebGates.
orainst.loc	Text file containing the location of your Oracle inventory, and also the group ID of the user creating the inventory.
webgate	This directory contains the Oracle HTTP Server Webgate binaries for Oracle Access Manager (OAM).
oui	This directory contains files used by the Oracle Universal Installer, including the uninstallation program. If you need to run the Oracle Universal Installer again after the product is installed, you should do so from this directory.
wlserver	This directory contains the WebLogic Server product files. This directory is also referred to as the WebLogic home directory, or <i>WL_HOME</i> .
domain-registry.xml	This registry file contains the locations of all domains created using the templates available in this Oracle home. Whenever you add a new domain, it is registered in this file.

# B

## Updating the JDK After Installing and Configuring an Oracle Fusion Middleware Product

Consider that you have an unsupported JDK version installed on your machine. When you install and configure an Oracle Fusion Middleware product, the utilities, such as Configuration Wizard (`config.sh|exe`), OPatch, or RCU point to a default JDK. The supported JDK version for this release is `jdk17.0.12` and it carries security enhancements and bug fixes. You can upgrade the existing JDK to a newer version, and can have the complete product stack point to the newer version of the JDK.

You can maintain multiple versions of JDK and switch to the required version on need basis.

### About Updating the JDK Location After Installing an Oracle Fusion Middleware Product

The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called `JAVA_HOME` which is centrally located in `.globalEnv.properties` file inside the `ORACLE_HOME/oui` directory.

The utility scripts such as `config.sh|cmd`, `launch.sh`, or `opatch` reside in the `ORACLE_HOME`, and when you invoke them, they refer to the `JAVA_HOME` variable located in `.globalEnv.properties` file. To point these scripts and utilities to the newer version of JDK, you must update the value of the `JAVA_HOME` variable in the `.globalEnv.properties` file by following the directions listed in [Updating the JDK Location in an Existing Oracle Home](#).

To make the scripts and files in your Domain home directory point to the newer version of the JDK, you can follow one of the following approaches:

- Specify the path to the newer JDK on the Domain Mode and JDK screen while running the Configuration Wizard.

For example, consider that you installed Oracle Fusion Middleware Infrastructure with the JDK version `8u191`. So, while configuring the WebLogic domain with the Configuration Assistant, you can select the path to the newer JDK on the Domain Mode and JDK screen of the Configuration Wizard. Example: `/scratch/jdk/jdk17.0.12`.

- Manually locate the files that have references to the JDK using `grep` (Linux) or `findstr` (WINDOWS) commands and update each reference.

See [Updating the JDK Location in an Existing Oracle Home](#).

#### Note:

If you install the newer version of the JDK in the same location as the existing JDK by overwriting the files, then you don't need to take any action.

## Updating the JDK Location in an Existing Oracle Home

The `getProperty.sh|cmd` script displays the value of a variable, such as `JAVA_HOME`, from the `.globalEnv.properties` file. The `setProperty.sh|cmd` script is used to set the value of variables, such as `OLD_JAVA_HOME` or `JAVA_HOME` that contain the locations of old and new JDKs in the `.globalEnv.properties` file.

The `getProperty.sh|cmd` and `setProperty.sh|cmd` scripts are located in the following location:

(Linux) `ORACLE_HOME/oui/bin`

(Windows) `ORACLE_HOME\oui\bin`

Where, `ORACLE_HOME` is the directory that contains the products using the current version of the JDK, such as `jdk17.0.12`.

To update the JDK location in the `.globalEnv.properties` file:

1. Use the `getProperty.sh|cmd` script to display the path of the current JDK from the `JAVA_HOME` variable. For example:

(Linux) `ORACLE_HOME/oui/bin/getProperty.sh JAVA_HOME`

(Windows) `ORACLE_HOME\oui\bin\getProperty.cmd JAVA_HOME`

`echo JAVA_HOME`

Where `JAVA_HOME` is the variable in the `.globalEnv.properties` file that contains the location of the JDK.

2. Back up the path of the current JDK to another variable such as `OLD_JAVA_HOME` in the `.globalEnv.properties` file by entering the following commands:

(Linux) `ORACLE_HOME/oui/bin/setProperty.sh -name OLD_JAVA_HOME -value specify_the_path_of_current_JDK`

(Windows) `ORACLE_HOME\oui\bin\setProperty.cmd -name OLD_JAVA_HOME -value specify_the_path_of_current_JDK`

This command creates a new variable called `OLD_JAVA_HOME` in the `.globalEnv.properties` file, with a value that you have specified.

3. Set the new location of the JDK in the `JAVA_HOME` variable of the `.globalEnv.properties` file, by entering the following commands:

(Linux) `ORACLE_HOME/oui/bin/setProperty.sh -name JAVA_HOME -value specify_the_location_of_new_JDK`

(Windows) `ORACLE_HOME\oui\bin\setProperty.cmd -name JAVA_HOME -value specify_the_location_of_new_JDK`

After you run this command, the `JAVA_HOME` variable in the `.globalEnv.properties` file now contains the path to the new JDK, such as `jdk17.0.12`.

## Updating the JDK Location in an Existing Domain Home

You must search the references to the current JDK, for example `1.8.0_191` manually, and replace those instances with the location of the new JDK.

You can use the `grep` or `findstr` commands to search for the JDK-related references.

You'll likely be required to update the location of JDK in the following three files:

(Linux) `DOMAIN_HOME/bin/setNMJavaHome.sh`

(Windows) `DOMAIN_HOME\bin\setNMJavaHome.cmd`

(Linux) `DOMAIN_HOME/nodemanager/nodemanager.properties`

(Windows) `DOMAIN_HOME\nodemanager\nodemanager.properties`

(Linux) **Start** `bash` **and then run** `DOMAIN_HOME/bin>source setDomainEnv.sh`

(Windows) `DOMAIN_HOME\bin\setDomainEnv.cmd`