Content Server

Version: 6.3

Installing Content Server with Oracle Application Server 10*g*

Document Revision Date: Dec. 1, 2005

FatWire SOFTWARE

Installing Content Server with Oracle Application Server 10g
Document Revision Date: Dec. 1, 2005
Product Version: 6.3

FatWire Technical Support
www.fatwire.com/Support

FatWire Headquarters
FatWire Corporation
330 Old Country Road
Suite 207
Mineola, NY 11501
www.fatwire.com

Table of

Contents

Part 3. Web Server

Part 4. Content Server

Appendixes

Chapter 1

Installation Overview

This document provides guidelines for installing Content Server on the Oracle Application Server 10*g*, connecting to a supported database of your choice.

Note

Anyone using this guide is expected to have experience installing and configuring databases, web servers, and application servers. Selected information regarding the configuration of third-party products is given in this guide. For detailed information about a particular third-party product, refer to that product's documentation.

This chapter provides information that will help you prepare for the Content Server installation. It contains the following sections:

- What This Guide Covers
- What This Guide Does Not Cover
- Installation Summary
- System Requirements
- Terms and Acronyms
- Graphics in This Guide
- Installation Steps

What This Guide Covers

This guide covers the installation, configuration, and maintenance of Oracle Application Server 10*g*, as required to support Content Server. This includes the configuration of an Oracle 10*g* cluster, backend databases, and standalone web servers.

What This Guide Does Not Cover

This guide does not cover the following topics, as they fall outside the scope of this guide:

- IIS and Sun ONE web server installation
- SSL configuration on IIS and Sun ONE
- LDAP integration

Installation Summary

After you install and configure the J2EE components that support Content Server, you will run the Content Server installer, which will guide you through the installation process. You will run the installer on each development, delivery, and management system on which you plan to use Content Server. During the Content Server installation, you will have the option to install or bypass sample sites and sample content, depending on the system you are setting up and on your business needs.

Note

The names of the systems in your Content Server environment might vary from the names used in this document. Generally, the management system is also called “staging,” and the delivery system is also called “production.”

System Requirements

System requirements for installing Content Server are given in the following documents, located on your Content Server installation CD:

- *Content Server Supported Platform List*. The list specifies third-party databases and drivers, application servers and web servers, and other software required for installing and running Content Server.
- *Content Server Release Notes*. The notes provide important information about Content Server.

FatWire recommends that you read both of these documents before installing Content Server.

Note

The latest versions of the above-mentioned documents are located at the following URL (password-protected):

`http://e-docs.fatwire.com/CS`

If you need a password, contact FatWire Technical Support. Contact information is available at the following URL:

`http://www.fatwire.com/Support/contact_info.html`

The e-docs website is organized by product and version number. To obtain the correct documents, follow the link for the version of Content Server you are installing.

Terms and Acronyms

The following table defines the terms and acronyms that are used throughout this guide.

Term	Definition
AS	Application Server
Oracle AS	Oracle Application Server 10*g*
OHS	Oracle HTTP Server
SSL	Secure Sockets Layer
Integrated OHS	Automatically installed Oracle HTTP Server (packaged with Oracle Application Server 10*g*).
Standalone OHS	Manually installed Oracle HTTP Server (packaged with Oracle Application Server 10*g*).

Graphics in This Guide

Many steps in this guide include screen captures of dialog boxes and similar windows that you interact with in order to complete the steps. The screen captures are presented to help you follow the installation process. They are not intended to be sources of specific information, such as parameter values, options to select, or product version number.

Installation Steps

The steps in this section summarize the installation/configuration of Content Server and its supporting software. Keep the steps handy as a quick reference to the installation procedure and to chapters that provide detailed instructions.

To install Content Server and its supporting software

1. Ensure that you have licensed copies of all the software you will be installing. For information about Content Server's supporting software, refer to the *Content Server Supported Platform List* and *Release Notes*. The latest versions are available on the e-docs website (password-protected), at the URL that is given in "System Requirements," on page 6.
2. Set up your choice of supported databases by installing the database management system, creating a database for Content Server, and configuring the database. For instructions, see our configuration guide, *Third-Party Software*.
3. Set up Oracle Application Server 10*g* as shown in Chapter 3, "Installing Oracle Application Server." The steps that you will complete are the following:
   a. Install Oracle Application Server 10*g*.
   b. Set up the environment and test the application server.
   c. If you plan to use SSL on the integrated (automatically installed) Oracle HTTP server (called "OHS" in this guide), configure SSL and create an SLL wallet. Otherwise, continue with the next step.
4. Configure Oracle Application Server, as shown in Chapter 4, "Configuring Oracle Application Server." The steps that you will complete are the following:
   a. Create and configure the data source, using either the command line or the graphical web-based method.
   b. If necessary, internationalize the database.
5. If you plan to install OHS manually (standalone OHS), or you prefer to install IIS or Sun ONE HTTP servers, see the following sections in Chapter 7, "Installing and Configuring a Web Server":
   - For instructions on installing OHS as a standalone instance, see "Installing and Configuring Oracle HTTP Server," on page 60.
   - For instructions on configuring OHS (both integrated and standalone), see "Configuring OHS (Integrated and Standalone)," on page 61.
   - If you plan to use SSL, follow instructions in "SSL (Optional)," on page 61 and "Creating an SSL Wallet (Integrated and Standalone)," on page 62.
   - For instructions on installing and configuring IIS, see "Configuring IIS Remote Plugin," on page 66.
   - For instructions on installing and configuring Sun ONE HTTP server, see "Configuring Sun ONE Remote Plugin," on page 67.
6. Install Content Server by running the installer. Halfway through the installation, you will need to deploy Content Server using either the command line or the graphical web-based method. For instructions on installing and deploying Content Server, see Chapter 8, "Installing Content Server."
7. If you plan to use the Verity search engine, follow installation guidelines in Appendix B, "Installing Verity Search Engine."
8. If you plan to set up a clustered installation, follow instructions in Chapter 3, "Installing Oracle Application Server."

Part 1

Database

This part contains a short chapter summarizing the databases that Content Server uses. Instructions on creating and configuring the databases are given our configuration guide, *Third-Party Software.*

This part contains the following chapter:

- Chapter 2, “Setting Up a Database”

Chapter 2

Setting Up a Database

Content Server requires access to a supported database that is specifically configured for Content Server. Supported databases for this release include:

- Oracle 9
- Oracle 10*g*
- Microsoft SQL Server 2000 SP3 and SP4
- DB2

The complete list of supported databases (as well as other third-party components) is given in the *Supported Platform List* (accessible from `http://e-docs.fatwire.com/CS`).

Before installing any other of Content Server's supporting software, you must complete the following steps:

1. Install the database management system.
   For instructions, refer to the product vendor's documentation.
2. Create and configure a database for Content Server.
   For instructions, consult our configuration guide, *Third-Party Software*. Note that database configuration is identical across different application servers. Refer to the correct chapter to create and configure the database of your choice.

Part 2

Application Server

This part contains information about installing and configuring Oracle Application Server. It also contains information about the functions that are performed with Oracle Application Server: application deployment and setting up clustered installations.

This part contains the following chapters:

- Chapter 3, “Installing Oracle Application Server”
- Chapter 4, “Configuring Oracle Application Server”
- Chapter 5, “Deploying Applications”
- Chapter 6, “Setting Up a Clustered Installation”

Chapter 3

Installing Oracle Application Server

The chapter shows you how to install Oracle Application Server. This is not an exhaustive chapter, as its covers the installation of Oracle Application Server (known throughout this manual as Oracle AS) only so far as needed to install and run Content Server. For more extensive documentation on the installation process, see the documentation that comes with Oracle Application Server.

This chapter contains the following sections:

- Pre-Installation Steps
- Installation Steps
- Post-Installation Steps

Pre-Installation Steps

The steps in this section must be completed before you can begin the Oracle AS installation. Failure to complete the steps will result in a failed installation attempt.

1. Create a new user (for example: `oracleas`).

   Unix creates a new user account named `oracleas` (you may choose any name that you wish, but in this guide we assume that you are using a user named `oracleas`).

2. Install any patches found in `<oracle install directory>\utils` directory. The directory varies according to platform, but must always be done as root or Administrator.

   Note

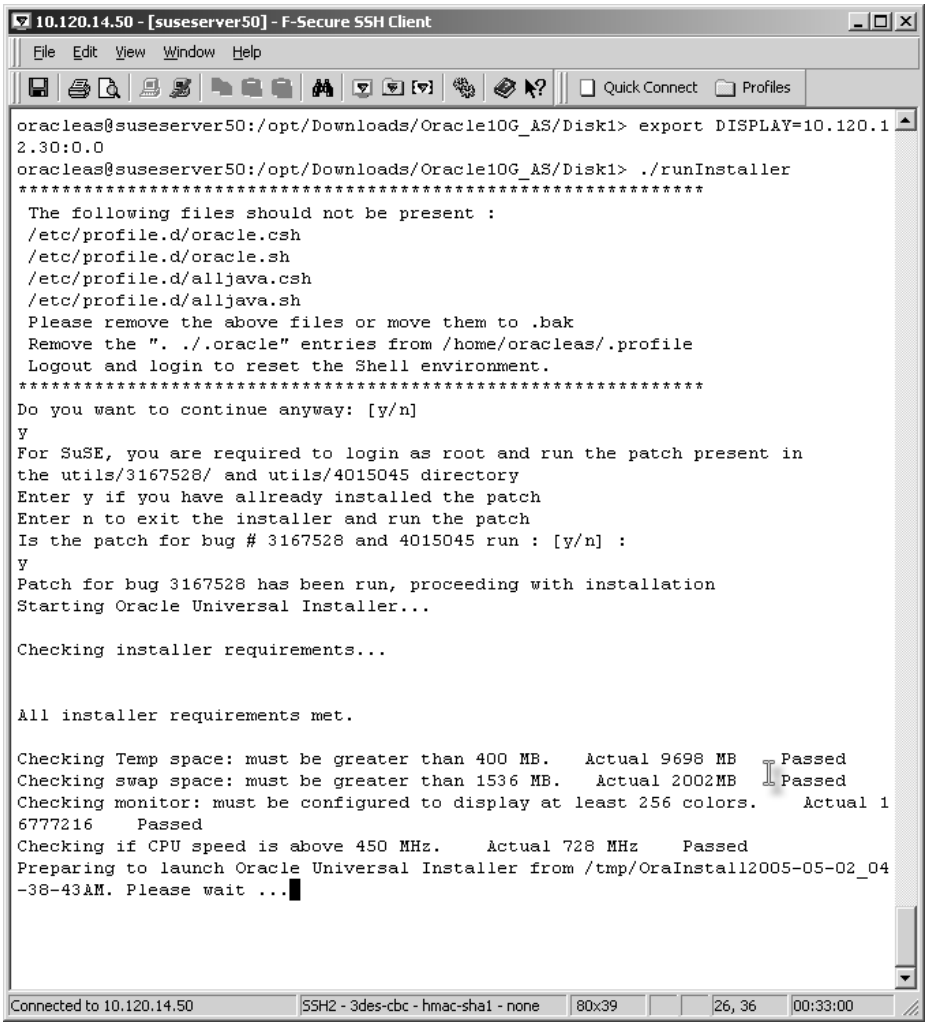
   In the following steps, we assume a Linux platform.

3. Log in as root:
   a. Change the directory to `<oracle install directory>/utils/3167528/` and execute the command: **`perl commentport.pl`**
   b. Change the directory to `<oracle install directory>/utils/4015045/` and execute the command: **`perl commentipv6.pl`**
4. Install any required patches for your given operating system. Information on required patches can be found in the release notes in `<oracle install directory>/docs`. If you are using SuSE Linux, install the required packages:
   a. Install the following packages using `yast2: openmotif, pdksh all packages`, and `gnomelibs`.
   b. When prompted about dependencies, install all dependent packages.
5. Change to the new user that you created in step 1. (In this guide the new user is `oracleas`).
6. Create the directory where you want Oracle AS to be installed (in this guide: `/opt/software/Apps/oracle10/oracleas/`). Ensure that the `oracleas` user created above is the owner of this directory.

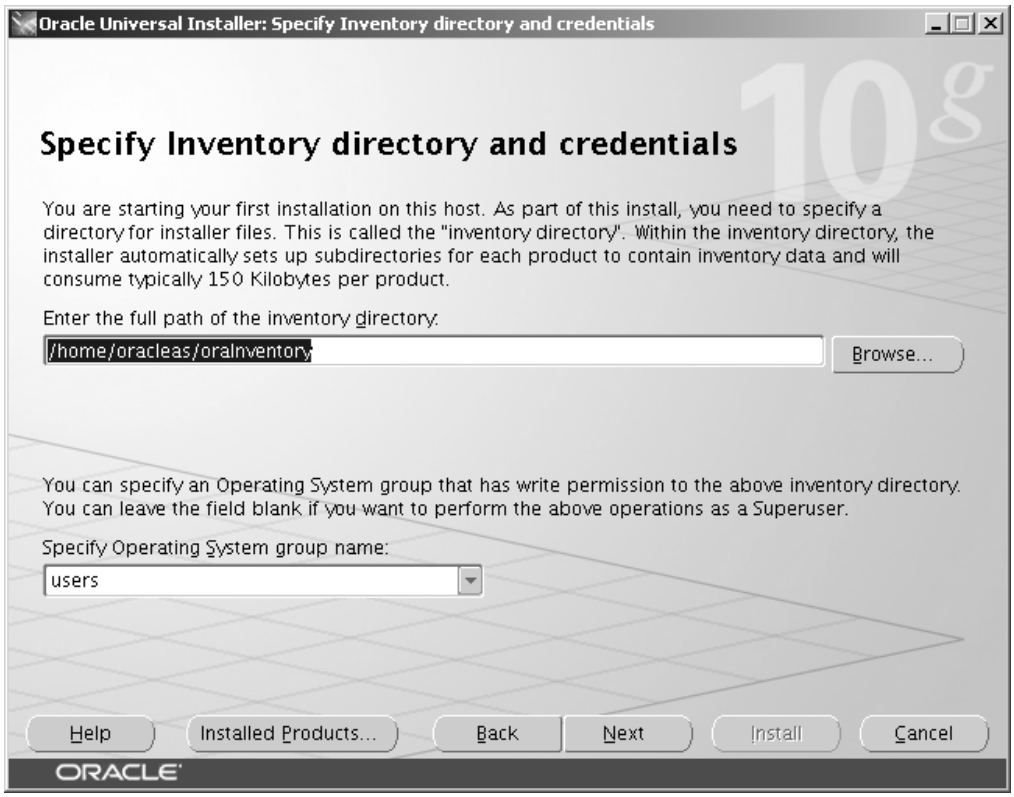
Installation Steps

This section steps you through the installation of the Oracle Application Server.

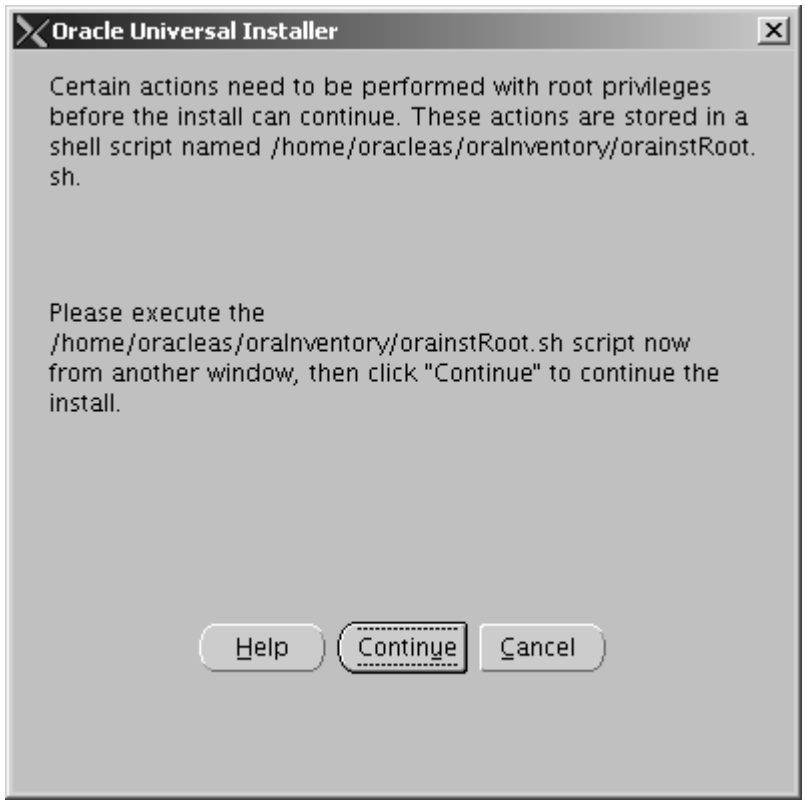
1. Run the installer, change to the `<oracle install directory>/`
   - On Unix: **`./runinstaller`**
   - On Windows: **`runinstaller.cmd`**



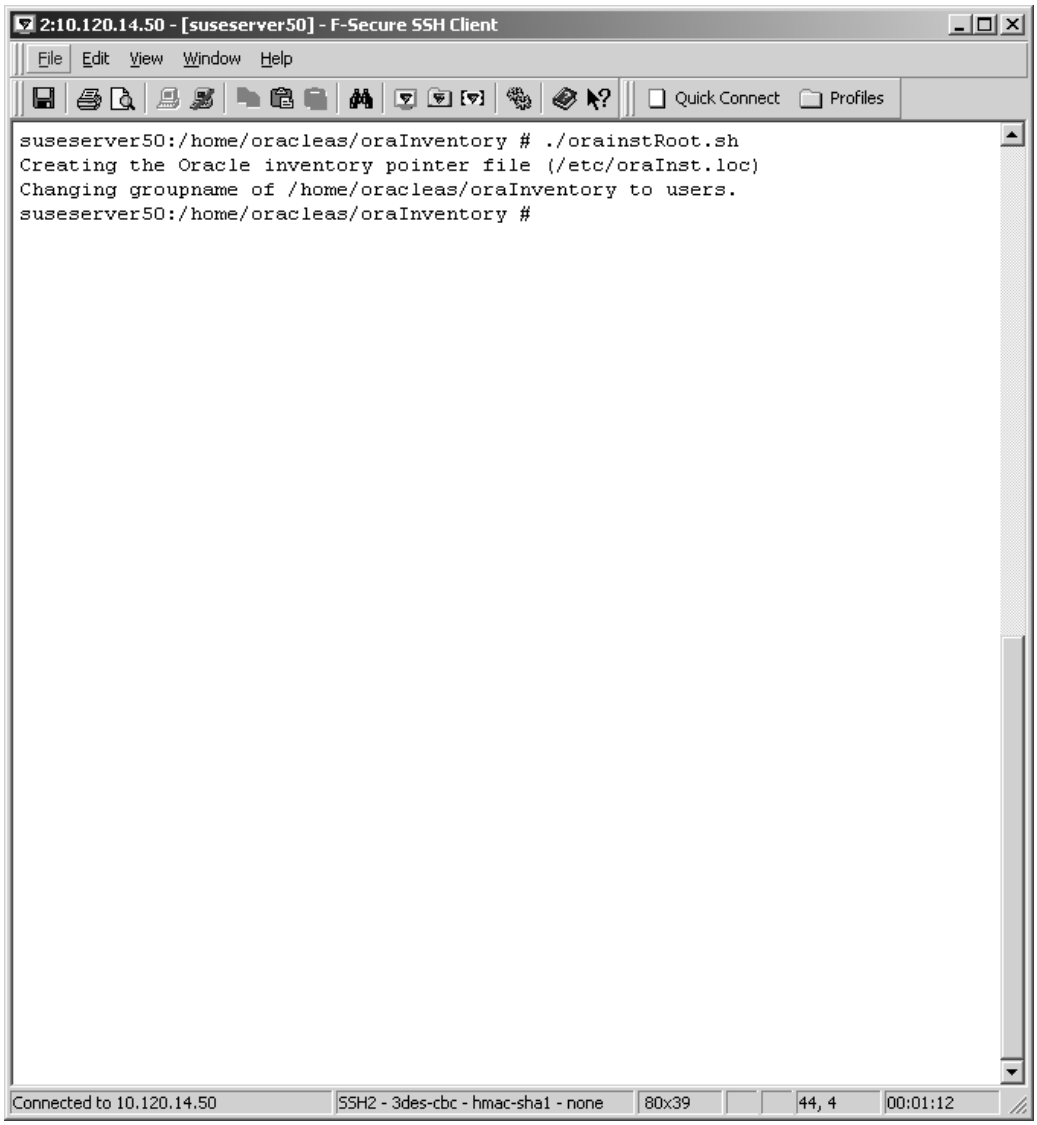
2. At the “Welcome” screen, click **Next**.
3. Set the inventory directory (use defaults) and click **Next**.



4. **For Unix only:**
   a. At the warning screen, **do not** click **Continue**. Instead go to step b on page 20.

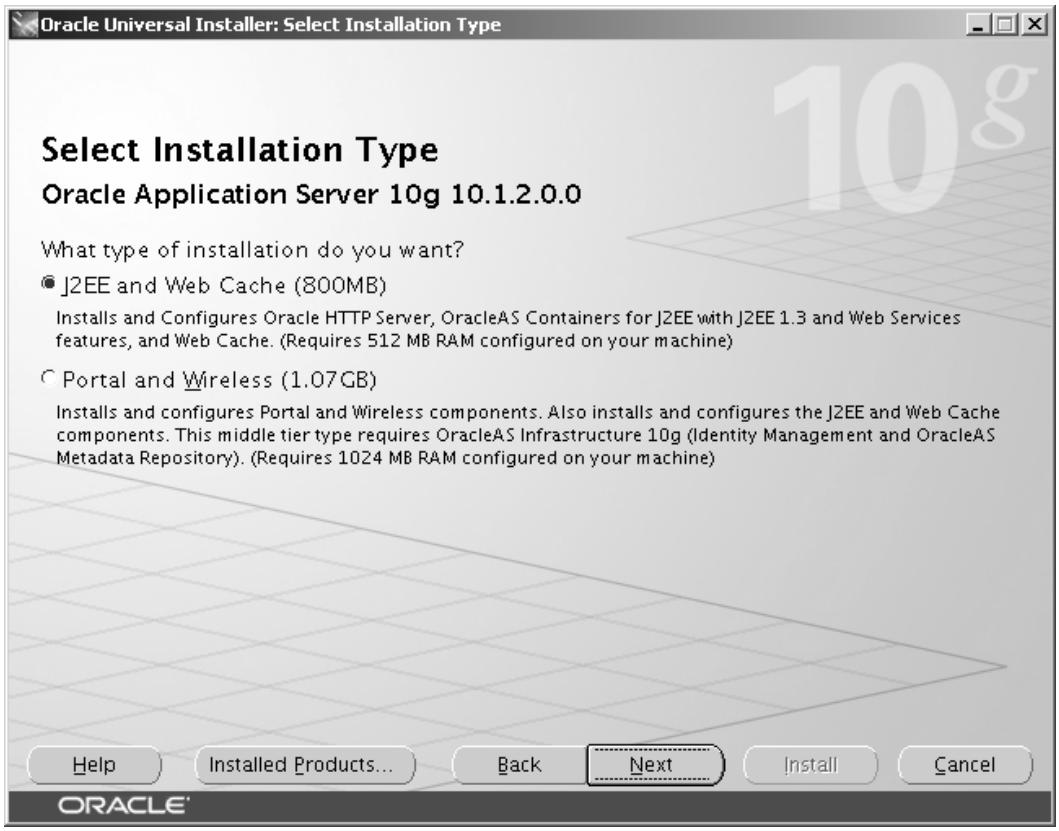


b. **For Unix only**: At a command prompt, log in as root, execute the script referred to in the pop-up message, then click **Continue** in the warning pop-up.

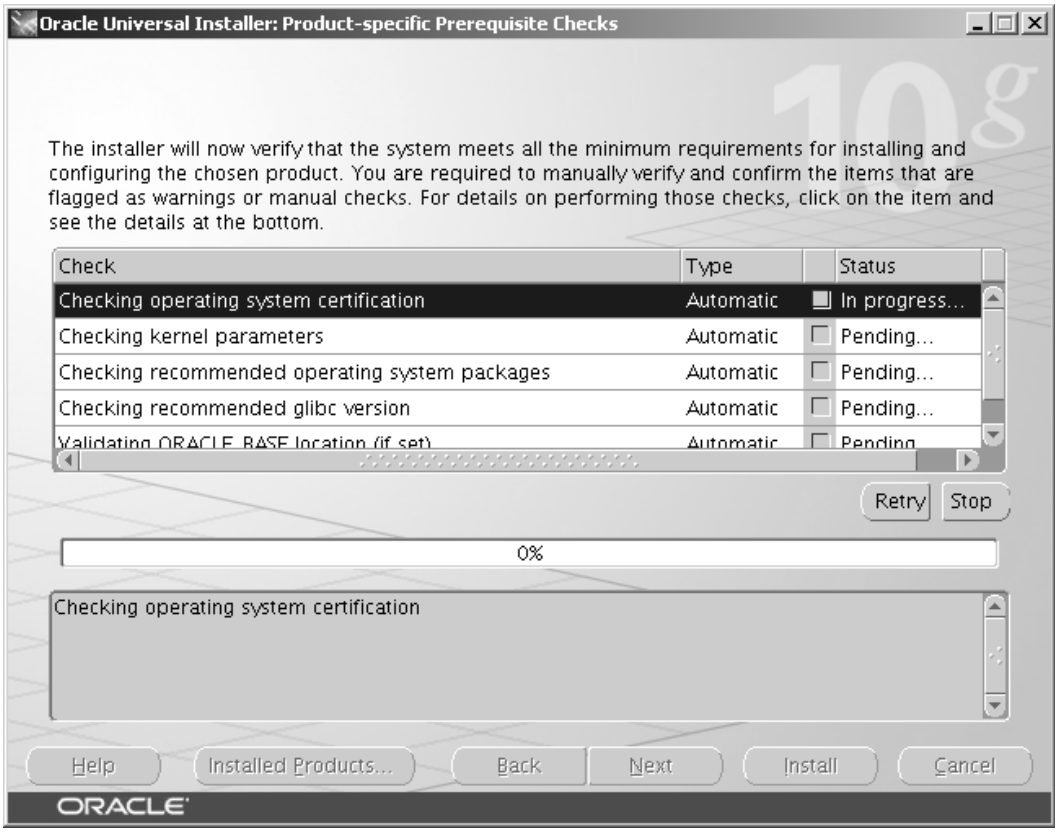


5. **For all operating systems**: Enter the location for the installation and click **Next**. This is the location that was created in the pre-installation step 6 on page 16.
6. Select the product to install (**Oracle Application Server 10*g***) and click **Next**.

7. Select the installation type (**J2EE and Web Cache**) and click **Next**.



8. **Perform pre-requisite checks:**
   a. Allow all checks to complete. If any checks fail, you will need to resolve them before continuing.
   b. Click **Next**.

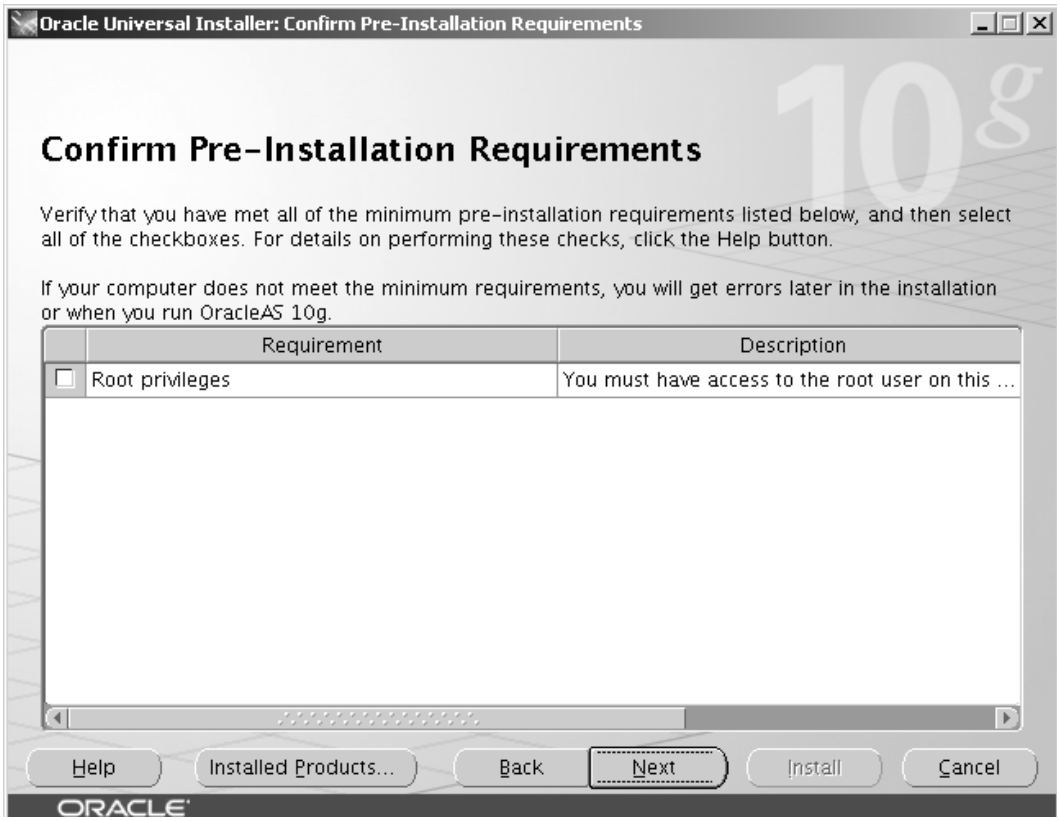


9. Confirm pre-installation requirements:

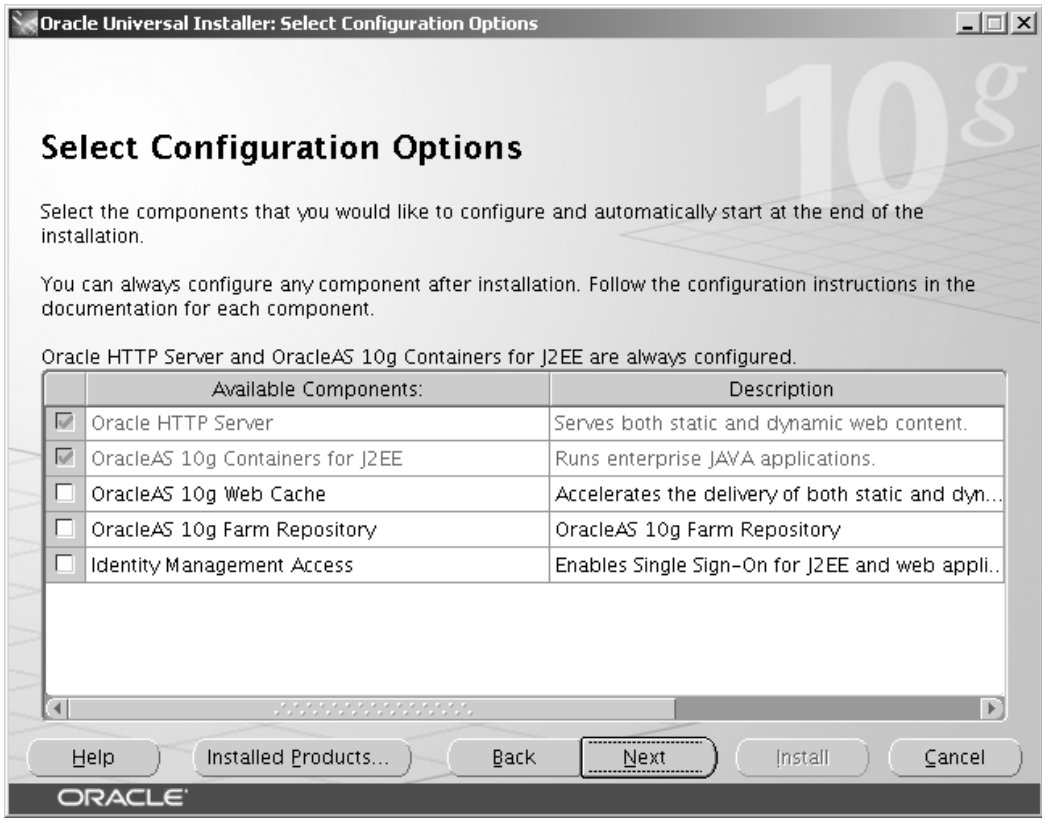
This screen contains a list of all requirements that must be met, but could not be tested for.

a. Place a check mark next to each item as you confirm that it is met as a requirement.

b. When all requirements have been met, click **Next**.



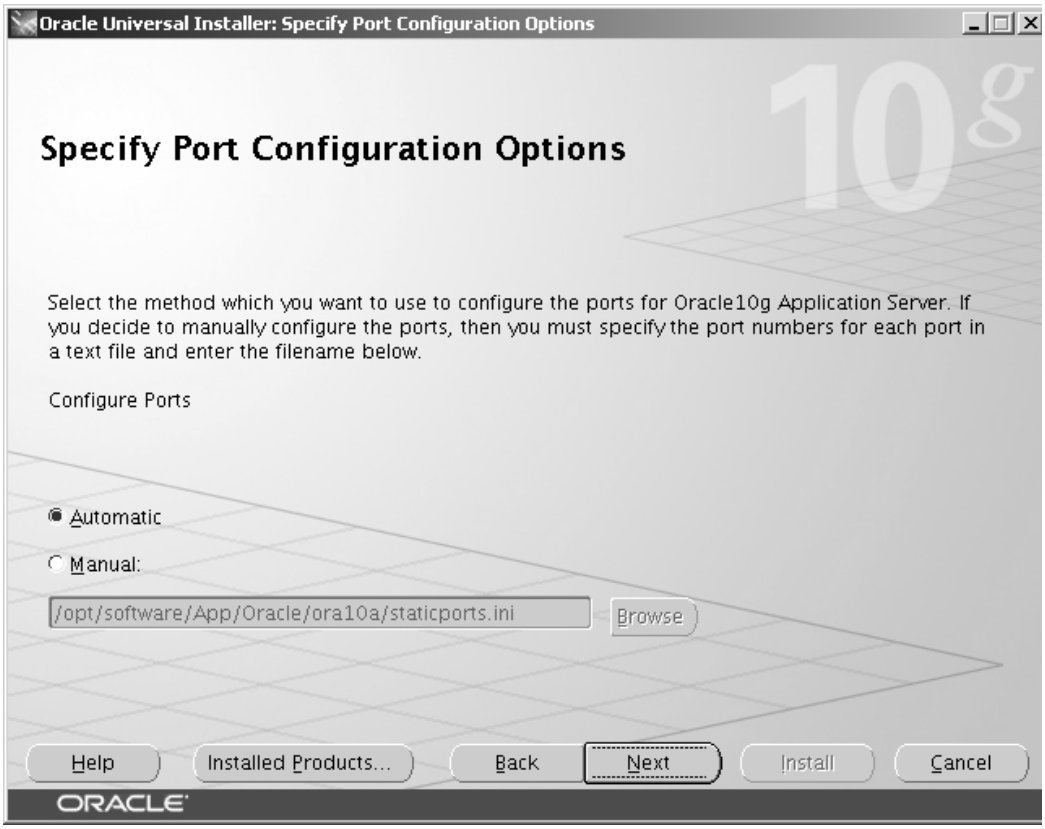
10. **Select configuration options:**
    a. Either leave the defaults selected or include **Oracle 10g Web Cache** if required. **Do not** select **Identity Manager** or **Farm Repository**.
    b. Click **Next**.



11. Specify port configuration options:

a. Leave the default **Automatic** selected.

b. Click **Next**.



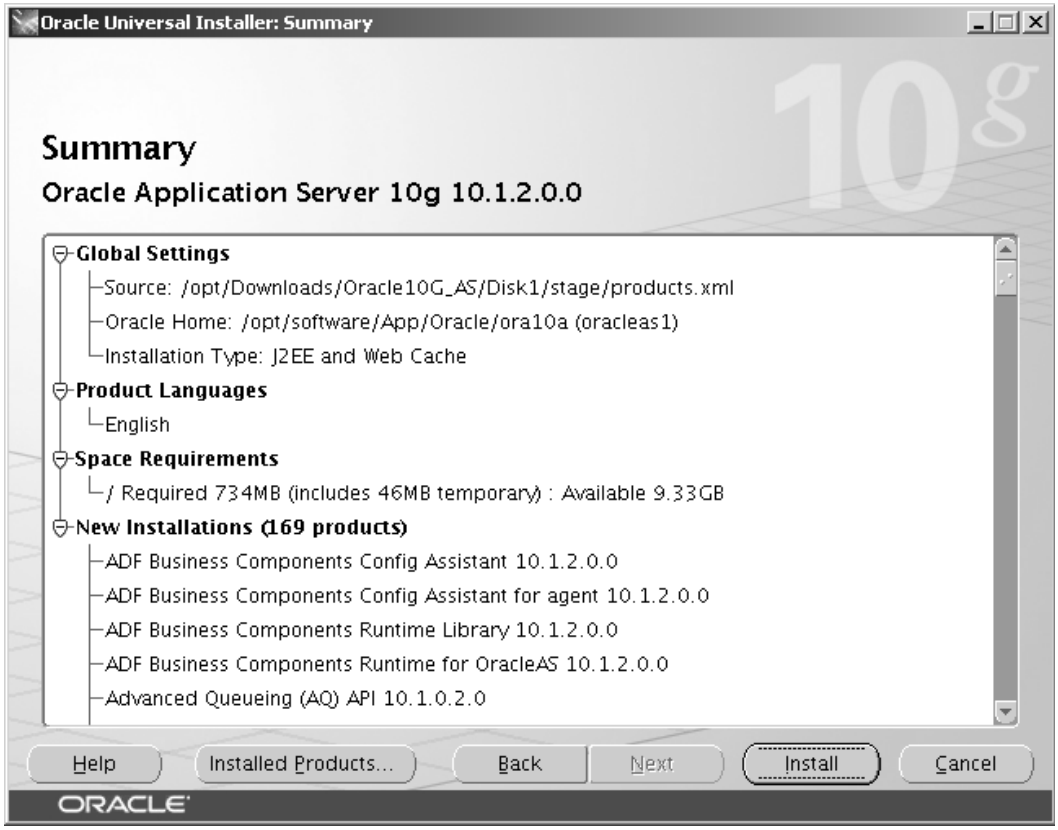
12. Specify instance name and password:

Enter a name and password for this instance and click **Next**. (In this guide, an instance name of `oracleas001` is used.)

> **Note**
>
> Keep the password in a safe place. If you forget this password, you will be unable to properly use Oracle Application Server.

13. Review the summary page of what will be installed:

Review the "Summary" page for any mistakes. If there are none, click **Install**.

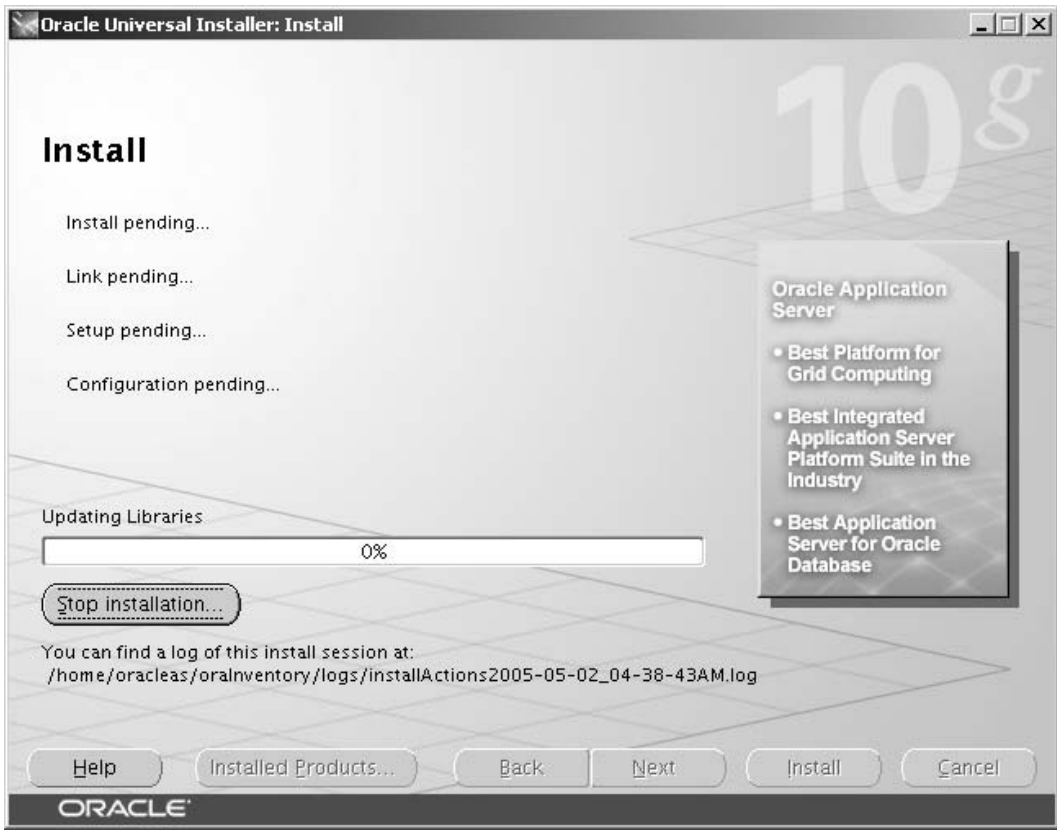


14. Installation progress screen:

a. Allow the installation process to complete.

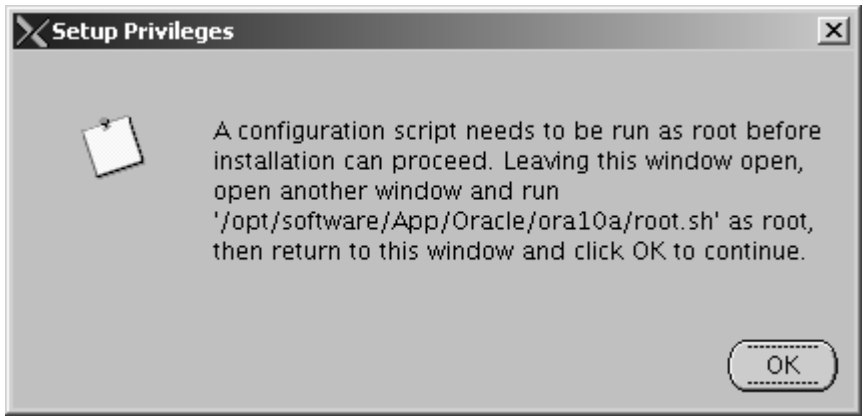
> **Note**
>
> Under Linux, the installation will fail with a message that opmn failed to start. The cause of this error is that a required Oracle library is not found.
>
> 1. Log in as root.
> 2. Add the directory `<ora home>/lib to /etc/ld.so.conf`.
> 3. Run `ldconfig`.
> 4. Execute the command: **`<ora home>/opmn/bin/opmn startall`**

b. Click **Next**.

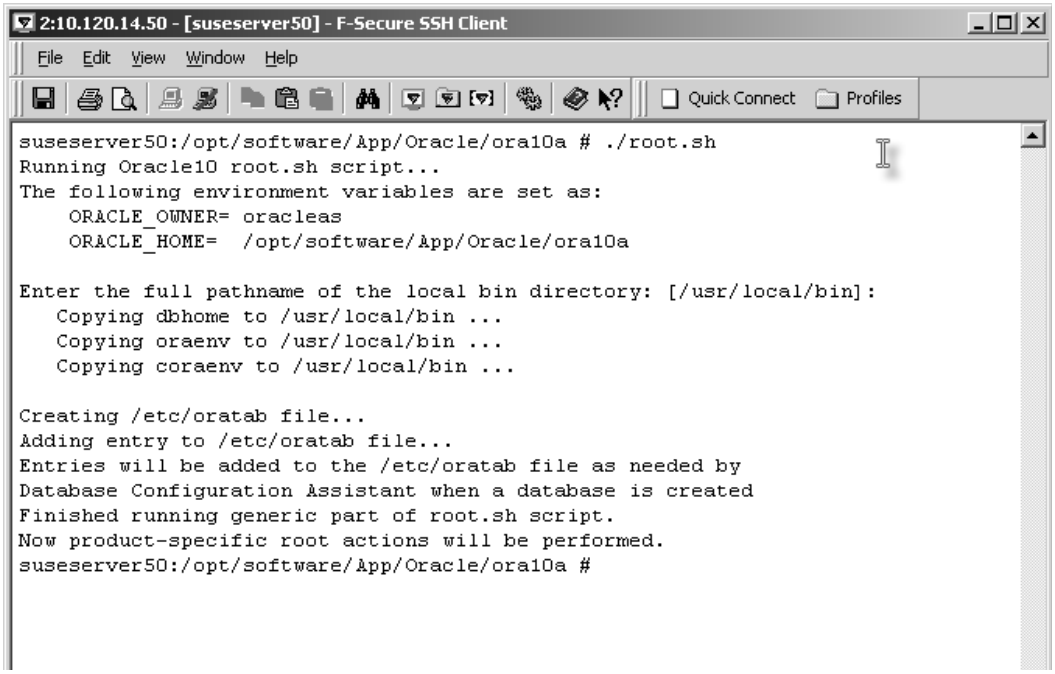


15. For Unix only:

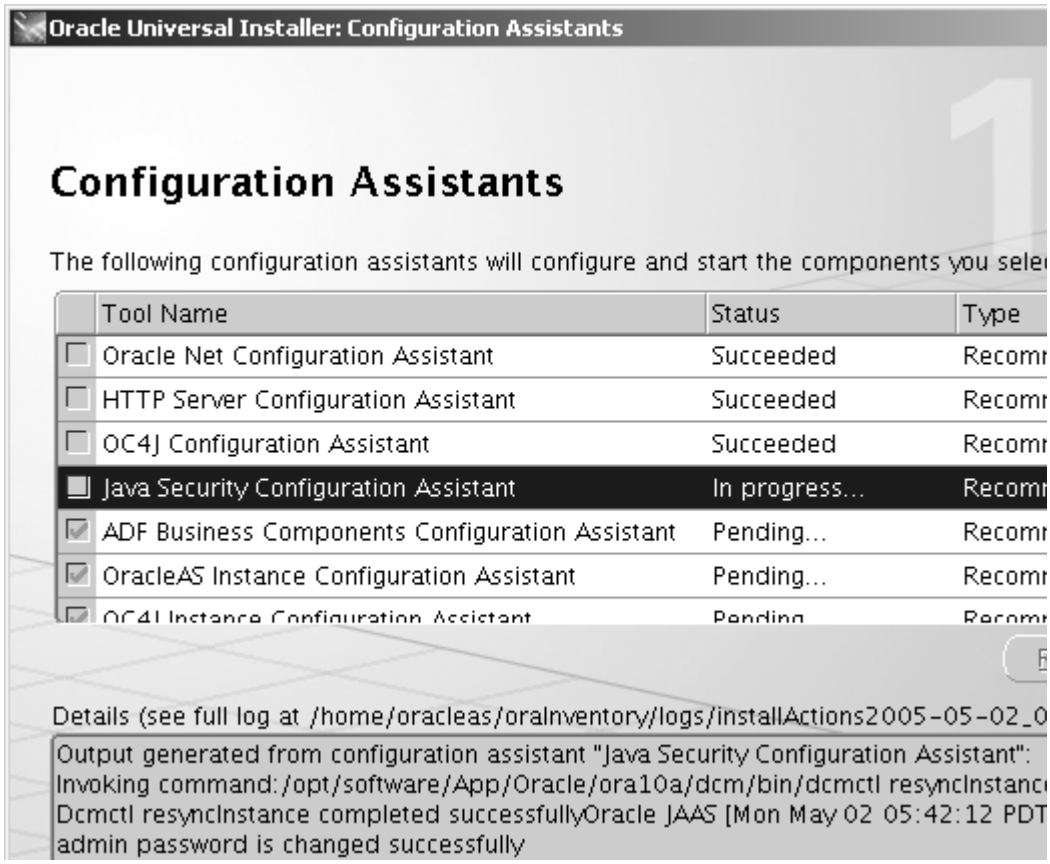
a. Follow instructions in the pop-up message.



b. At a command prompt, log in as root, execute the script referred to in the pop-up message, then click **OK** in the pop-up.



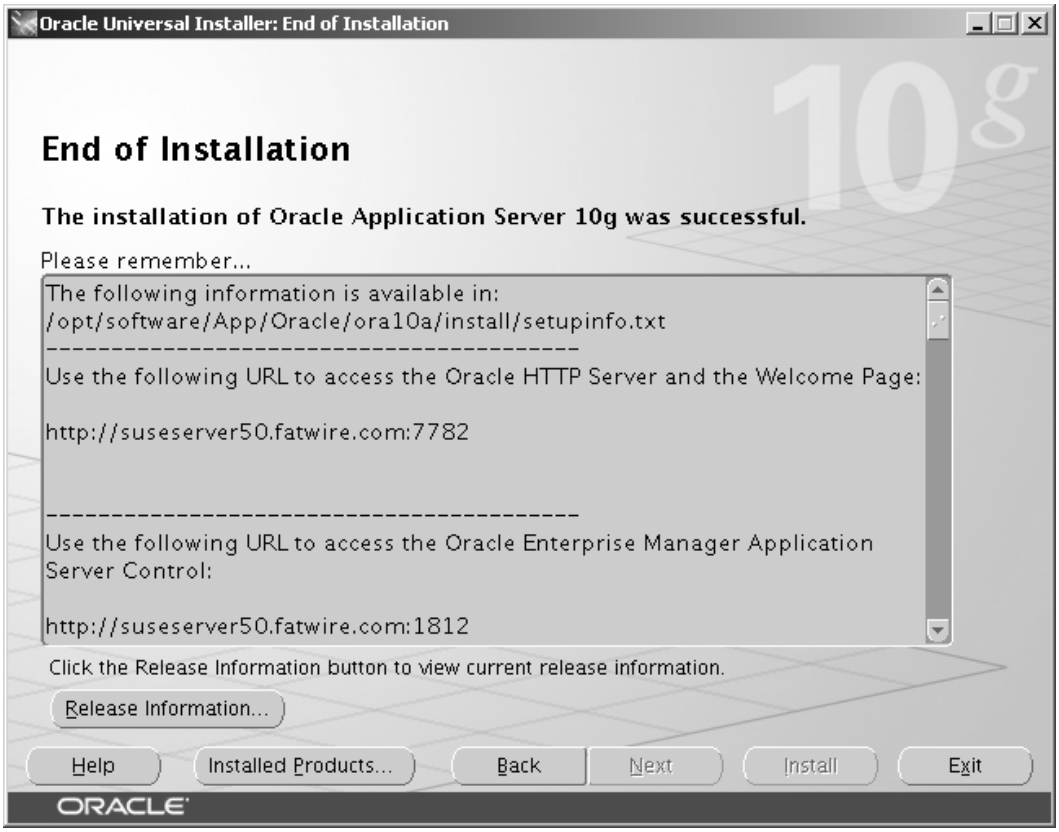
16. **For all operating systems:**
    a. Allow all the Oracle configuration assistants to finish successfully.
    b. Click **Next**.



17. The Oracle AS installation is now complete.

a. Write down the URL displayed on the screen for the Oracle Enterprise Manager Application (normally the host name and port 1812).

b. Click **Exit**.



Post-Installation Steps

A. Set Up and Test Your Environment

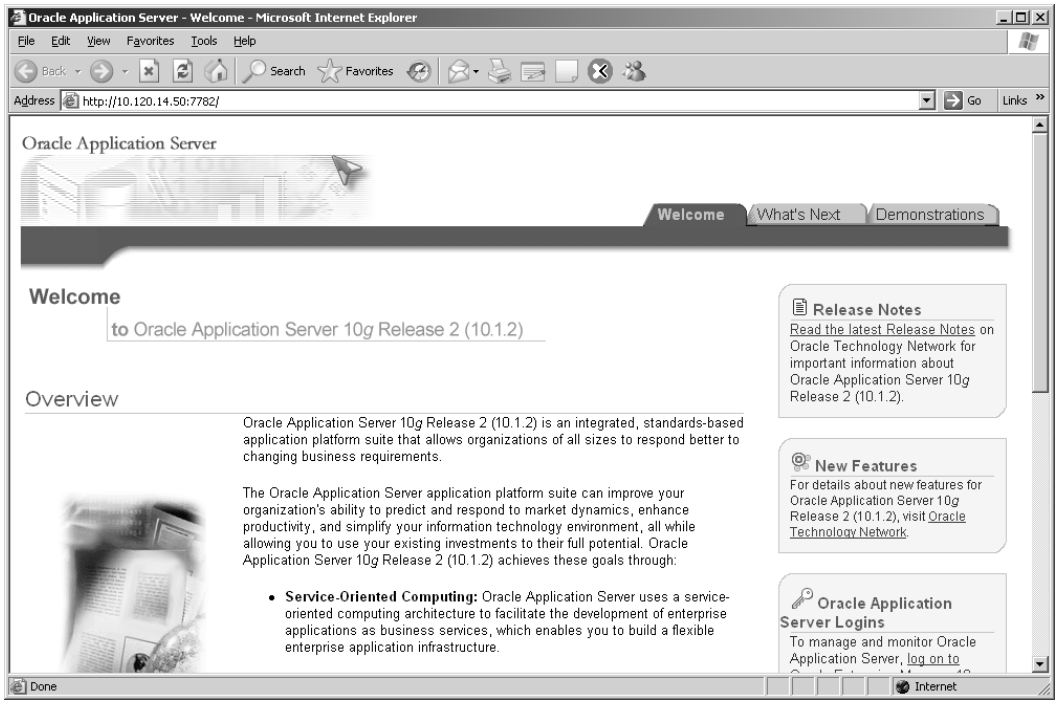
1. If you are using Unix, edit `.profile` for your Oracle AS user (`oracleas` in this guide) by adding the following lines to the end of the file:

```
EXPORT PATH=<ora home>/jre/1.4.2/bin:$PATH
EXPORT ORACLE_HOME=<ora home>/
```

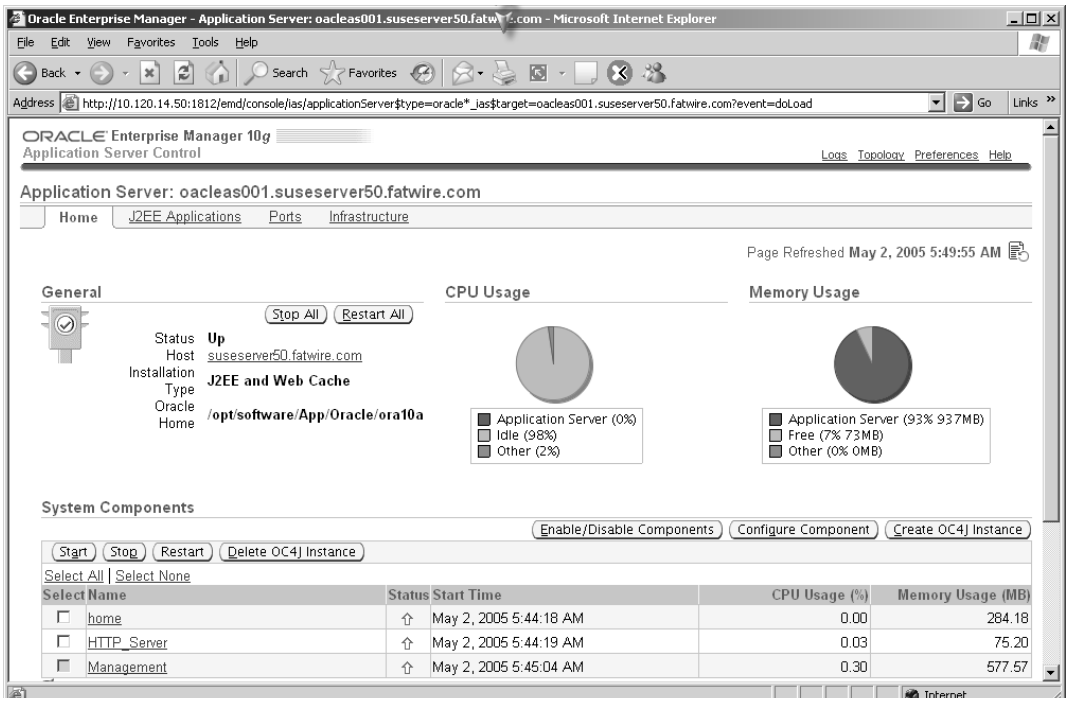
> **Note**
>
> Throughout the rest of this guide `<ora home>` refers to the path where Oracle AS was installed.

2. Save the file and re-source the file . **`~/.profile.`**

3. Add the following directories to your library path: `<ora home>/lib` and `<ora home>/chgip/lib`
4. Open a browser and browse to the URL that was provided on the last screen during the installation process (step 17 on page 32).



5. Log in to Oracle Application Server using the login **ias_admin** and the password that was entered during the installation process (step 12 on page 27).



6. You have completed installing and testing Oracle Application Server. Your next step is one of the following:
   - If you plan to use SSL on the integrated Oracle HTTP server, continue with section B.
   - If you are not planning to use SSL on the integrated Oracle HTTP Server, configure the Oracle application server. For instructions, see Chapter 4, "Configuring Oracle Application Server," in particular, "Data Source Creation and Configuration," on page 38.

B. If You Plan to Use SSL on the Integrated Oracle HTTP Server

1. If you plan to use the integrated (automatically installed) Oracle HTTP Server, enable SSL by following instructions in "SSL (Optional)," on page 61 and in "Creating an SSL Wallet (Integrated and Standalone)," on page 62.
2. When you have completed step 1, configure the Oracle application server. For instructions, see Chapter 4, "Configuring Oracle Application Server," in particular, "Data Source Creation and Configuration," on page 38.

Chapter 4

Configuring Oracle Application Server

This chapter shows you how to construct a data source and how to configure Oracle Application Server to support internationalization as well as clustered installations. For reference, this chapter begins with a section on basic Oracle AS information that is used throughout this guide and is required for configuring and maintaining Content Server.

This chapter contains the following sections:

- Basic Information and Operations
- Data Source Creation and Configuration
- Database Internationalization

Basic Information and Operations

This section contains basic Oracle AS information that is required for configuring and maintaining Content Server. Many of the files and commands that are described in this section will be used throughout the rest of this guide. However, only the basics are covered here. Consult the Oracle Application Server product documentation for more extensive information on the topics that are covered in this guide and for topics that are not touched upon.

Important Files and Their Locations

File	Description	Path
`<application name>`	Per-application log	`<ora home>/j2ee/<instance name>/application-deployments/<application name>`
`data-sources.xml`	Used for viewing and modifying globally available data sources	`<ora home>/j2ee/<instance name>/config/data-sources.xml`
`data-sources.xml`	Used for viewing and modifying application-specific available data sources	`<ora home>/j2ee/<instance name>/application-deployments/<app name>/data-sources.xml`
`emiasconsole.nohup`	Enterprise Management Console log	`<ora home>/sysman/log/emiasconsole.nohup`
`portlist.ini`	Used for managing the ports on which Oracle AS is currently configured to listen on	`<ora home>/install/portlist.ini`
`setupinfo.txt`	Used for viewing and editing the ports on which Oracle AS was configured to listen on during installation	`<ora home>/install/setupinfo.txt`
deployed applications path		`<ora_home>/j2ee/<instance name>/applications/<application name>`
third-party jar files, installation path		`<ora home>/j2ee/<instance name>/applib/`
dcm (Oracle Distributed Configuration Manager) configuration files		`<ora home>/dcm/config/`
dcm (Oracle Distributed Configuration Manager) logs		`<ora home>/dcm/logs/`
opmn (Oracle Process Manager and Notification Server) and configuration files		`<ora home>/opmn/conf/`

File	Description	Path
opmn (Oracle Process Manager and Notification Server) application logs		`<ora home>/opmn/logs/`

Important Commands

> **Note**
>
> For all commands it is assumed that `ORACLE_HOME` is set and that you are logged in as the Oracle Application Server user (`oracleas` in this guide).

Table 1: Oracle Enterprise manager Web Console

Action	Command
Start	**<ora home>/bin/emctl start iasconsole**
Stop	**<ora home>/bin/emctl stop iasconsole**
Status	**<ora home>/bin/emctl status iasconsole**

Table 2: Oracle Application Server

Action	Command
Start all	**opmnctl startall**
Stop all	**opmnctl stopall**
Status of Oracle AS components	**opmnctl status**
Start a single component	**opmnctl startproc ias-component=<name>**
Stop a single component	**opmnctl stopproc ias-component=<name>**

Application Deployment Methods

Oracle Application Server allows for an application to be deployed either from the command line or through the graphical interface. The command line is preferred because deployment from the Oracle Enterprise Manager requires the EAR file to be on the same physical machine as the browser. Instructions for application deployment are given in Chapter 5, “Deploying Applications.”

Data Source Creation and Configuration

In order for Content Server (or any application) to communicate with a database, you must first construct a data source. This section discusses the creation and configuration of an Oracle AS data source that is capable of communicating with a supported database.

Oracle AS supports two ways of creating a data source: 1) through the graphical console, and 2) by direct editing of an xml file. Both methods are covered in this section. However, the graphical method is suggested unless you have expertise with editing xml files. Again, this chapter covers only those points that are necessary to the installation of Content Server. For a more thorough explanation of data sources or for clarification of something you do not understand, consult the documentation that comes with Oracle AS.

XML-Based Procedure

1. To add a new data source through an xml file, ensure that the opmn instance onto which you will add this data source is shut down.
2. Back up the file `data-sources.xml` (the path is `<ora home>/j2ee/<instance name>/config/data-sources.xml`)
3. Edit the file `<ora home>/j2ee/<instance name>/config/data-sources.xml`

   By default this file contains a single data source. However, depending on your server configuration this file might contain more than one data source.

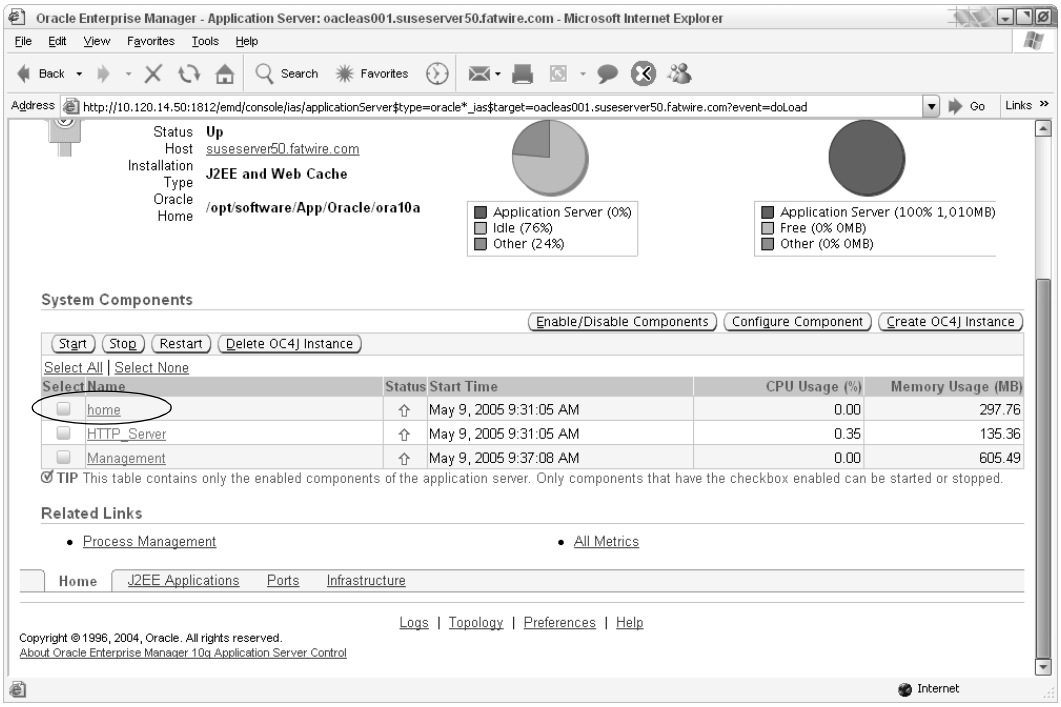
   At the end of the file is the following string: `</data-sources>`
   Immediately before this string, insert a new data source. The easiest way is to copy the example below:

   ```
   <data-source class="com.evermind.sql.DriverManagerDataSource"
     name="<display name>" location="jdbc/<name of datasource>"
     xa-location="jdbc/<name of datasource>1" ejb-location="jdbc/
     <name of datasource>2" connection-driver="<driver name>"
     username="<user name>" password="<user password>"  url="<DB
     connection URL>"  inactivity-timeout="30" />
   ```
4. Once you have copied the information above, change the values of the following variables in `data-sources.xml`:

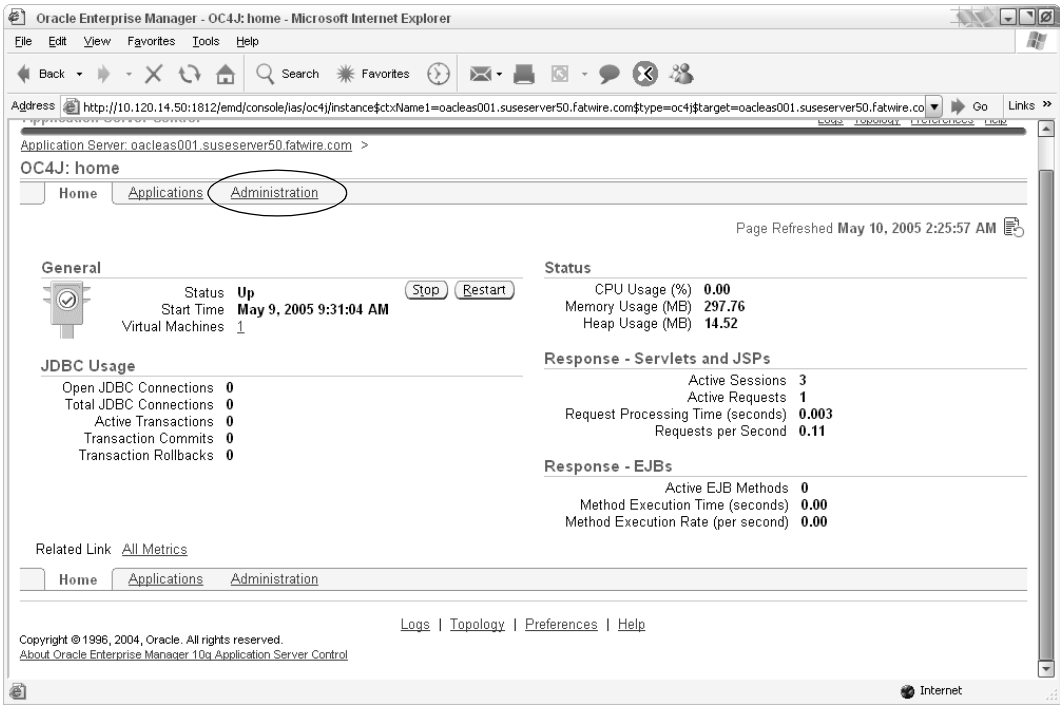
Variable	Description
`<display name>`	The display name of the data source.
`<name of datasource>`	The name to use with applications to access this data source.
`<driver name>`	The name of the driver to use to connect to your given database type. See below for common formats.
`<user name>`	User name to access this database.
`<user password>`	User's password to access this database.
`<DB connection URL>`	URL to use to connect to this database, see common formats below.

Web-Based Procedure

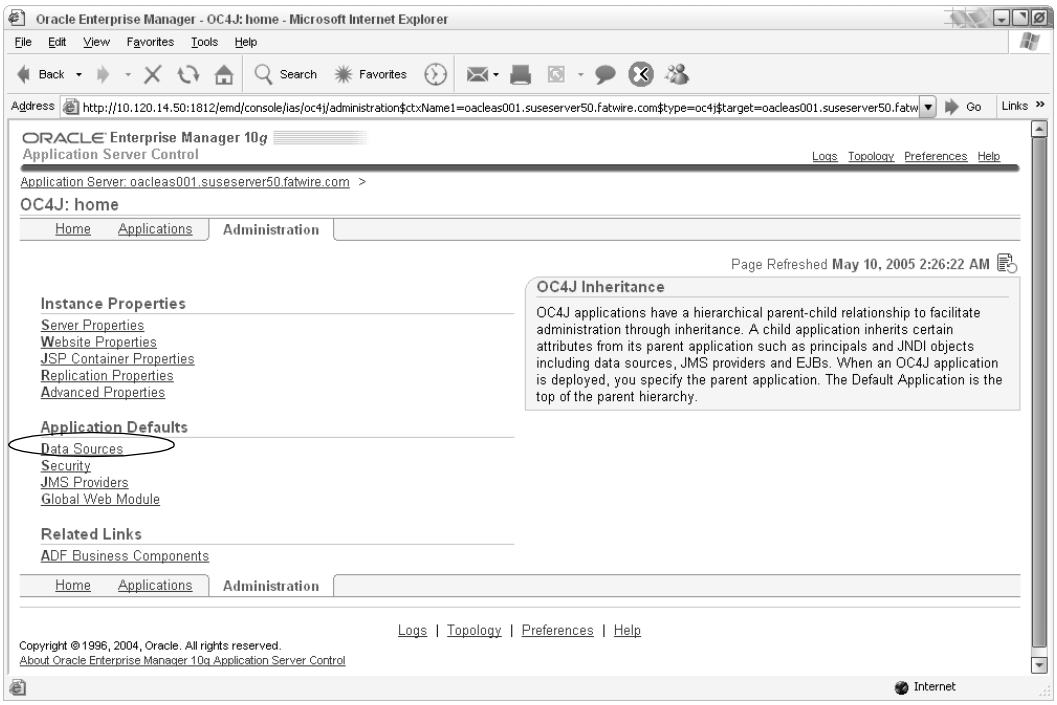
1. In the "Oracle Enterprise Manager," select the instance onto which to add the data source and click its name (in this example, **home**).



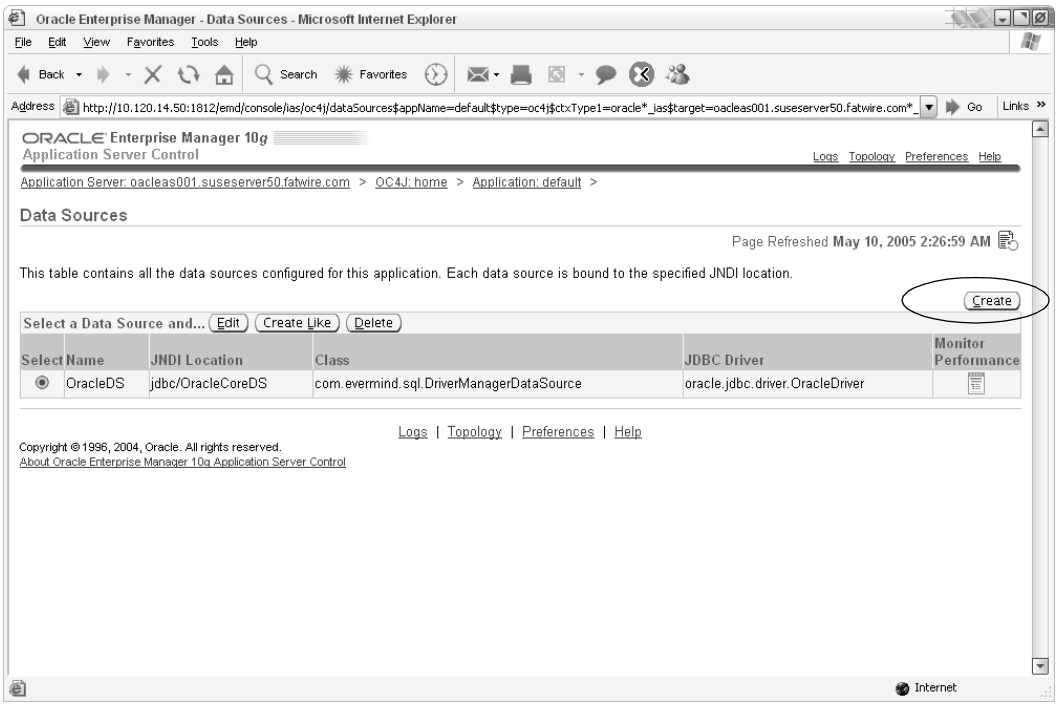
2. From the instance page, click the link named **Administration**.



3. On the "Administration" page, click the link **Data Sources**.



4. Click the **Create** button.



5. Fill in the form (using the table below for guidelines) and click **Create**.

Field	Description
Name	The name you want the data source to be displayed as. Value: `<display name>`
Data Source Class	The name to use with applications to access this data source. Value: `<name of datasource>`

JDBC URL	URL to use to connect to this database.
	Value: `<DB connection URL>`
	Common formats:
	• Oracle 9, 10 Driver URL Format:
	`jdbc:oracle:thin:@//<servername>:<port>/<db>`
	E.g.,
	`jdbc:oracle:thin:@//10.120.14.22:1521/CS52NP`
	• Microsoft SQL Server 2000 SP4 Driver URL Format:
	`jdbc:microsoft:sqlserver://<server name>:<ip>; SelectMethod=Cursor;DatabaseName=<DB>`
	E.g.,
	`jdbc:microsoft:sqlserver://10.120.14.22:1433; SelectMethod=Cursor;DatabaseName=CS62OA`
	• DB2 Driver URL Format:
	`jdbc:db2://<servername>:<port>/<DB>`
	E.g.,
	`jdbc:db2://10.120.16.30:50001/DB2OA`

JDBC Driver	Name of the driver to use to connect to your given database type.
	Value: `<driver name>`
	Common formats:
	• Oracle 9, 10 Driver Name:
	`oracle.jdbc.driver.OracleDriver`
	• Microsoft SQL Server 2000 SP4 Driver Name:
	`com.microsoft.jdbc.sqlserver.SQLServerDriver`
	• DB2 Driver Name:
	`com.ibm.db2.jcc.DB2Driver`

Username	User name to access this database. Value: `<user name>`
Use Cleartext Password	Value: `<user password>`
Password	User's password to access this database. Value: `<user password>`
Location	Value: `jdbc/<name of datasource>`

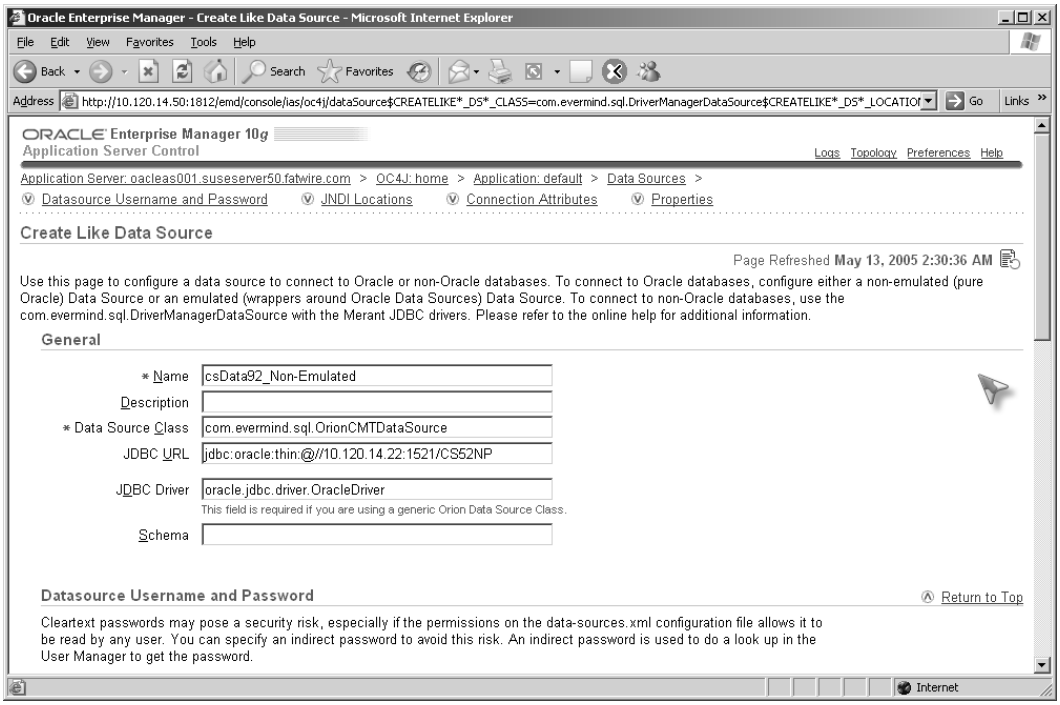
Field	Description
Transactional (XA) Location	Value: `jdbc/<name of datasource>`
EJB Location	Value: `jdbc/<name of datasource>`
Connection Retry Interval (Seconds)	Value: `1`
Cached Connection Inactivity Timeout (Seconds)	Value: `30`

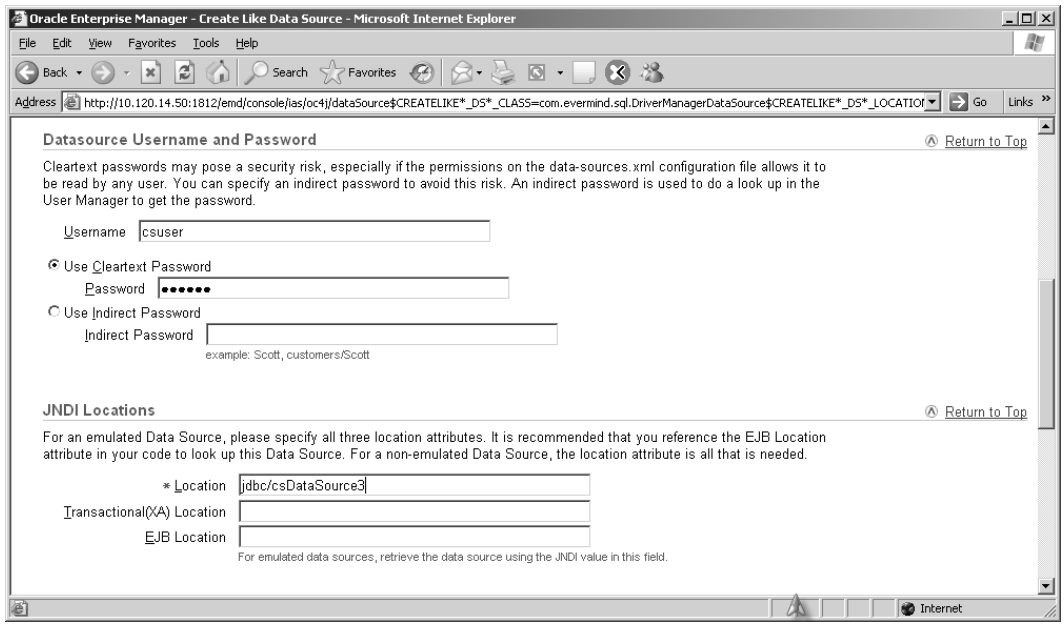
Note

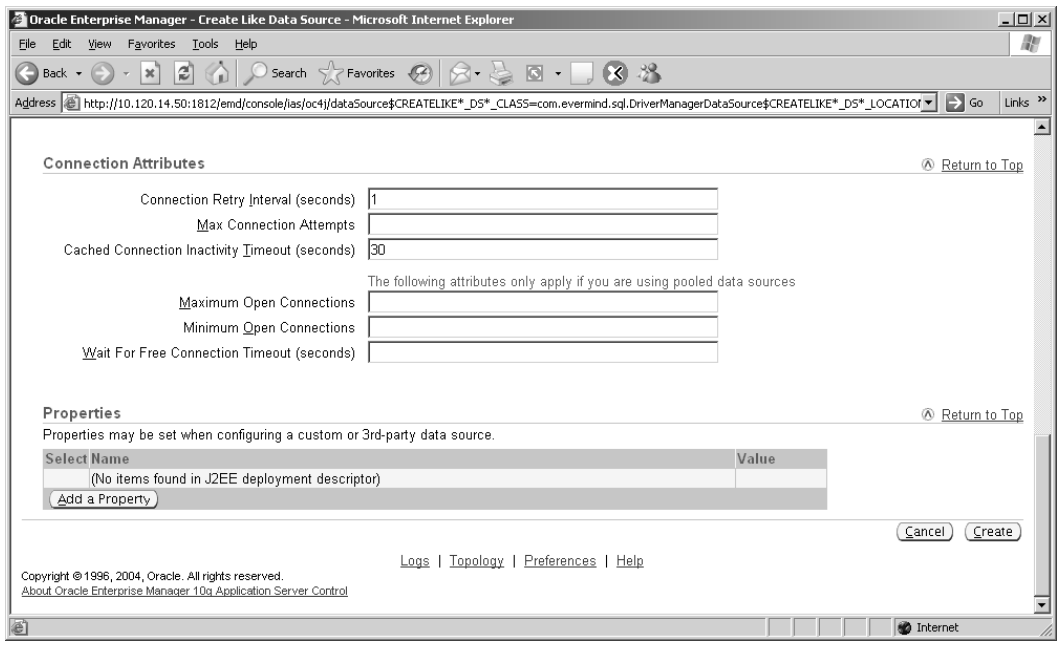
The following files are required third-party jar files that must be added to the directory `<ora home>/j2ee/<instance name>/applib/`:

Microsoft SQL Server 2000 SP4: `msutil.jar`, `mssqlserver.jar`, `msbase.jar`

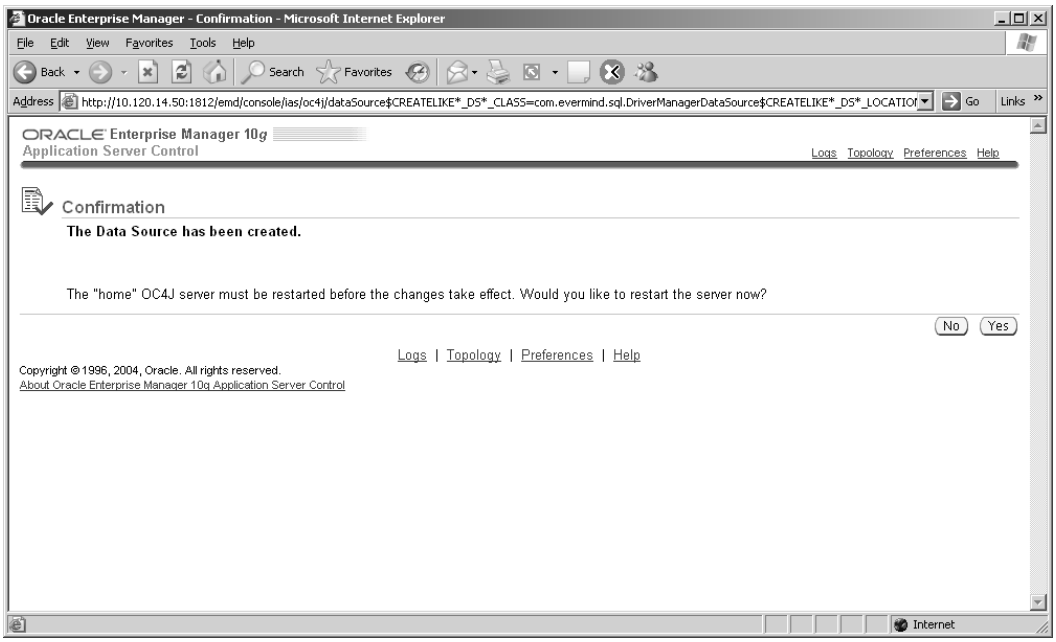
DB2: `db2jcc.jar`, `db2jcc_license_cu.jar`




Oracle Enterprise Manager - Create Like Data Source - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://10.120.14.50:1812/emd/console/ias/oc4j/dataSource$CREATELIKE*_DS*_CLASS=com.evermind.sql.DriverManagerDataSource$CREATELIKE*_DS*_LOCATIO
Go Links
Datasource Username and Password
Return to Top
Cleartext passwords may pose a security risk, especially if the permissions on the data-sources.xml configuration file allows it to be read by any user. You can specify an indirect password to avoid this risk. An indirect password is used to do a look up in the User Manager to get the password.
Username csuser
Use Cleartext Password
Password ••••••
Use Indirect Password
Indirect Password
example: Scott, customers/Scott
JNDI Locations
Return to Top
For an emulated Data Source, please specify all three location attributes. It is recommended that you reference the EJB Location attribute in your code to look up this Data Source. For a non-emulated Data Source, the location attribute is all that is needed.
* Location jdbc/csDataSource3
Transactional(XA) Location
EJB Location
For emulated data sources, retrieve the data source using the JNDI value in this field.
Internet


Oracle Enterprise Manager - Create Like Data Source - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://10.120.14.50:1812/emd/console/ias/oc4j/dataSource$CREATELIKE*_DS*_CLASS=com.evermind.sql.DriverManagerDataSource$CREATELIKE*_DS*_LOCATIO
Go Links
Connection Attributes
Return to Top
Connection Retry Interval (seconds) 1
Max Connection Attempts
Cached Connection Inactivity Timeout (seconds) 30
The following attributes only apply if you are using pooled data sources
Maximum Open Connections
Minimum Open Connections
Wait For Free Connection Timeout (seconds)
Properties
Return to Top
Properties may be set when configuring a custom or 3rd-party data source.
Select Name Value
(No items found in J2EE deployment descriptor)
Add a Property
Cancel Create
Logs | Topology | Preferences | Help
Copyright © 1996, 2004, Oracle. All rights reserved.
About Oracle Enterprise Manager 10g Application Server Control
Internet

6. When prompted to confirm restart, click **Yes**.



Database Internationalization

If you wish to utilize international (non-Western charters), complete the following steps before installing any applications:

1. Edit the files `opnmctl` (in `<ora home>/opmn/bin/`) and `apachectl (in <ora home>/Apache/Apache/bin/)` by changing the value of the `NLS_LANG` parameter to `"AMERICAN_AMERICA.UTF8"` (shown in bold type below):

   Original:
   `NLS_LANG=${`**`NLS_LANG="AMERICAN_AMERICA.WE8ISO8859P1"`**`}; export NLS_LANG`

   Change to:
   `NLS_LANG=${`**`NLS_LANG="AMERICAN_AMERICA.UTF8"`**`}; export NLS_LANG`

2. Restart all OPMN services:

   ```
   <ora home>\opmn\bin\opmnctl stopall
   <ora home>\opmn\bin\opmnctl startall
   ```

Chapter 5

Deploying Applications

This chapter covers the deployment of applications with Oracle Application Server, using the command line and the graphical interface.

This chapter contains the following sections:

- Overview
- Command Line Deployment (Preferred)
- Web-Based Deployment

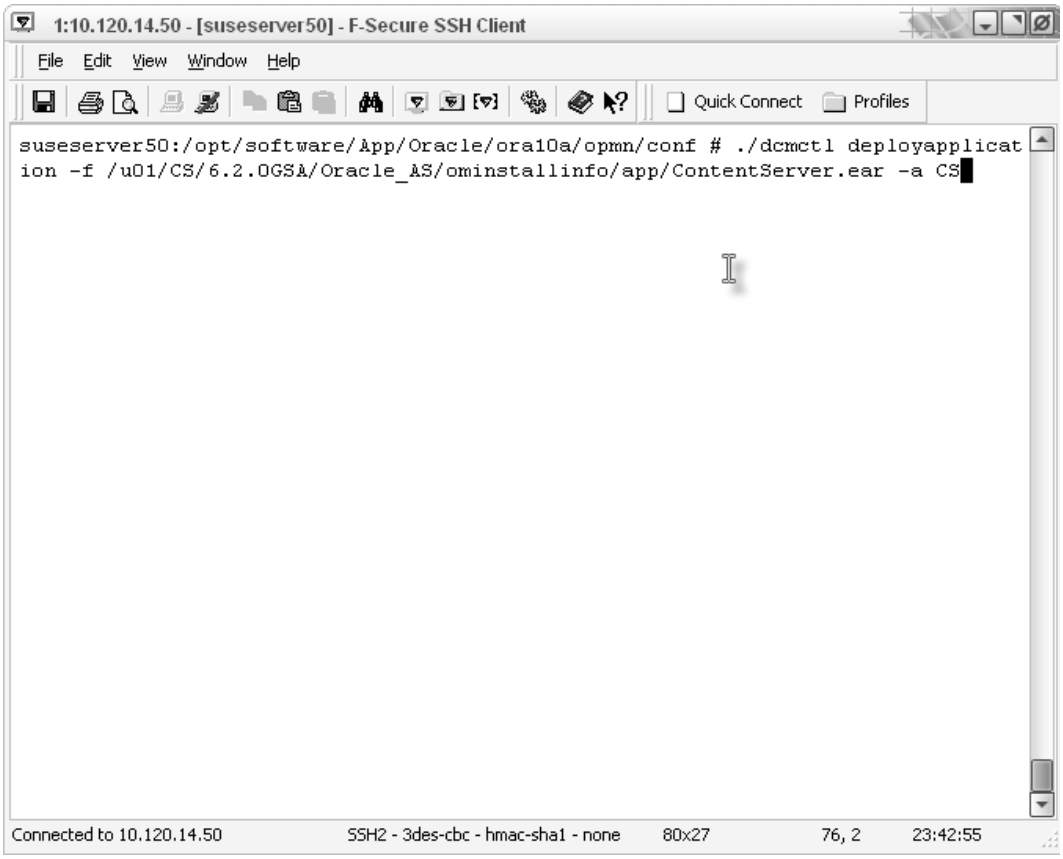
Overview

Oracle Application Server allows for an application to be deployed either from the command line or through the graphical interface. The command line is preferred because deployment from the Oracle Enterprise Manager requires the EAR file to be on the same physical machine as the browser. Both the command line method and graphical interface methods are given in this section.

Command Line Deployment (*Preferred*)

1. Log in as the Oracle User.
2. Change directories to `<ora home>\dcm\bin\`
3. Use the "Oracle Distributed Deployment Manager" to deploy an EAR file:

```
<ora home>/dcm/bin/dcmctl deployapplication \
-f /u01/CS/6.2.0GSA/Oracle_AS/ominstallinfo/app/
ContentServer.ear \
-a CS
```



4. Use the Oracle distributed deployment manager to undeploy an EAR file:

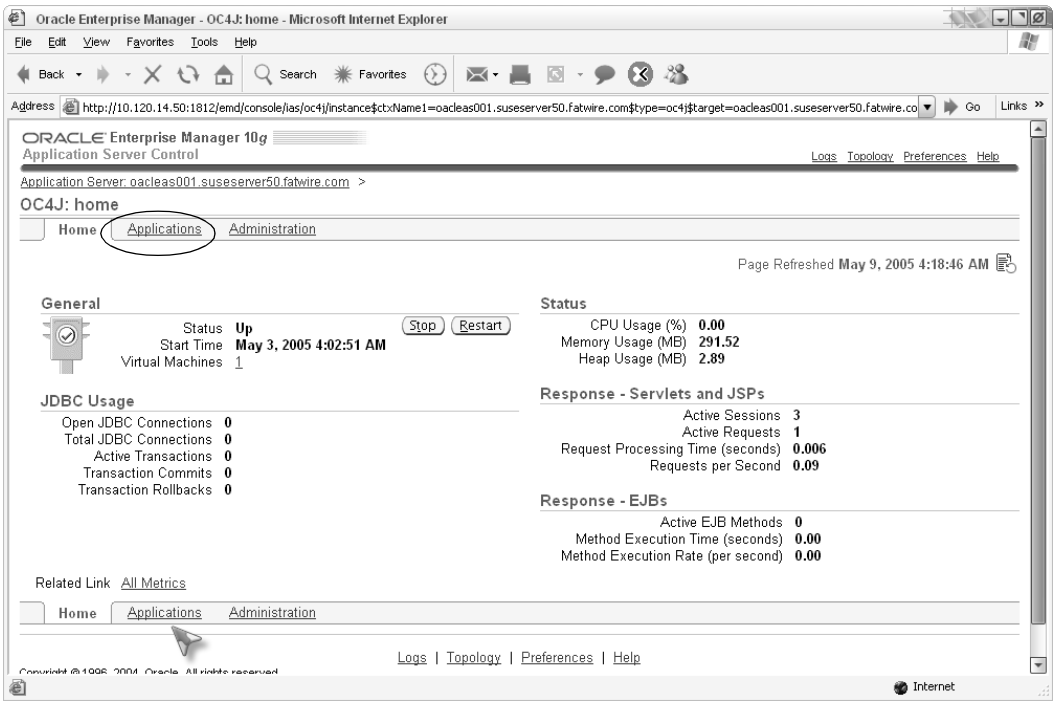
```
<ora home>/dcm/bin/dcmctl undeployapplication -a CS
<ora home>/dcm/bin/dcmctl resyncInstance
```

Web-Based Deployment

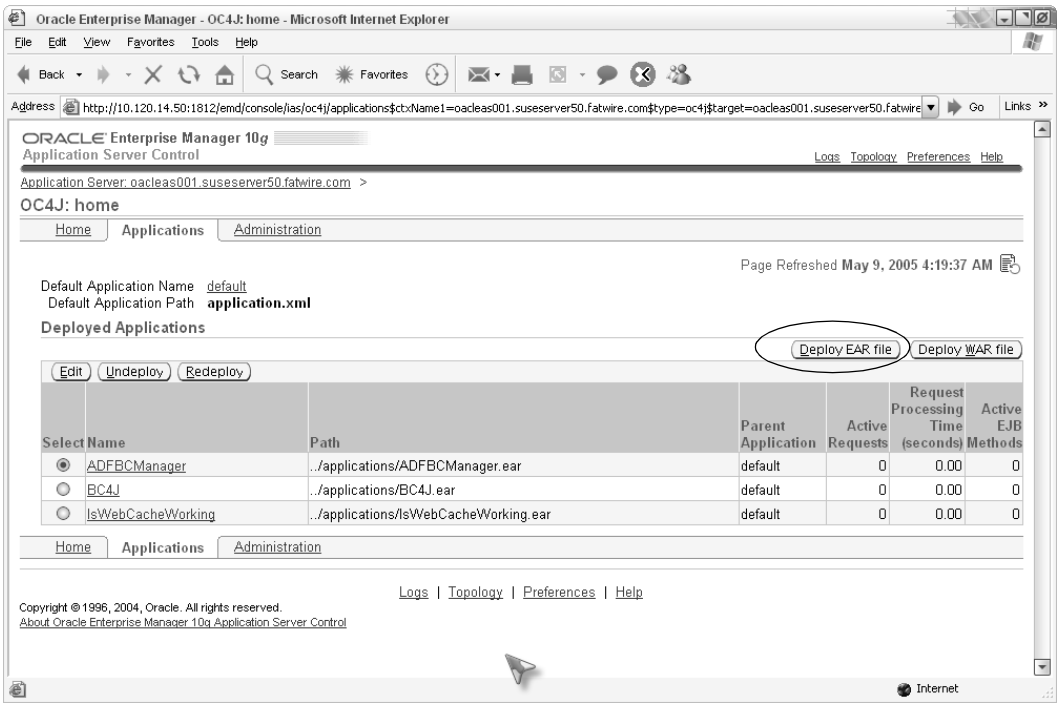
Note

For web-based deployment of an application, the EAR file must be local to the browser.

1. Log in to the "Oracle Enterprise Manager" and click **Applications**.



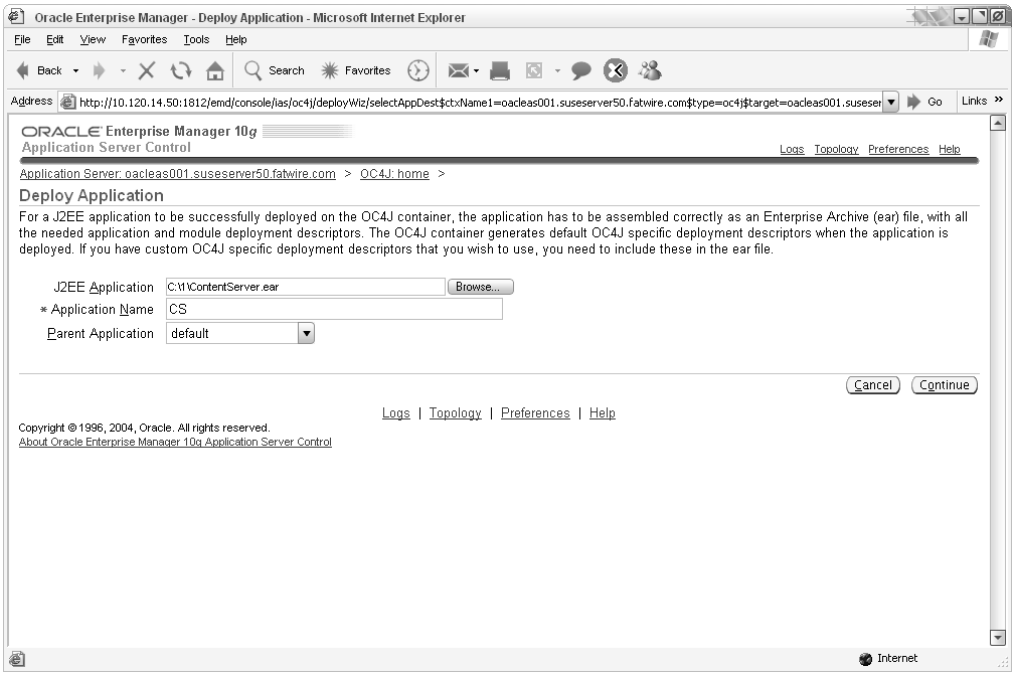
2. On the **Application**, click **Deploy EAR file**.



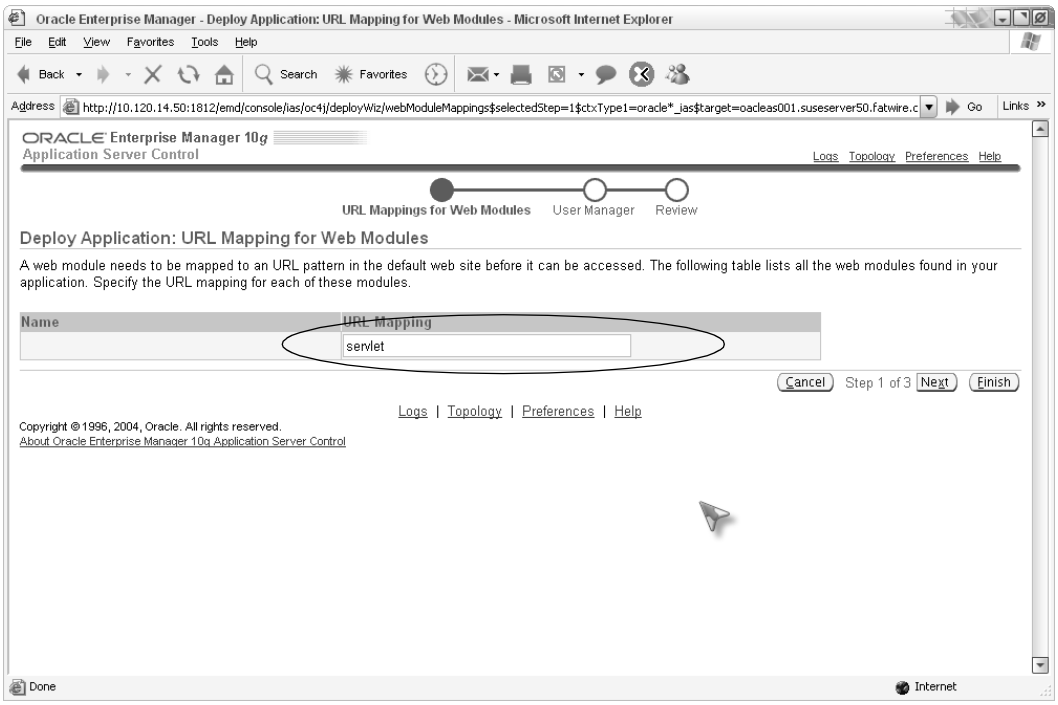
3. On the EAR file deployment page, fill out the form as follows:
   a. Enter the local location of the EAR file.

> **Note**
>
> In order to deploy using the web interface, the EAR file must be on the local file system of the browser.

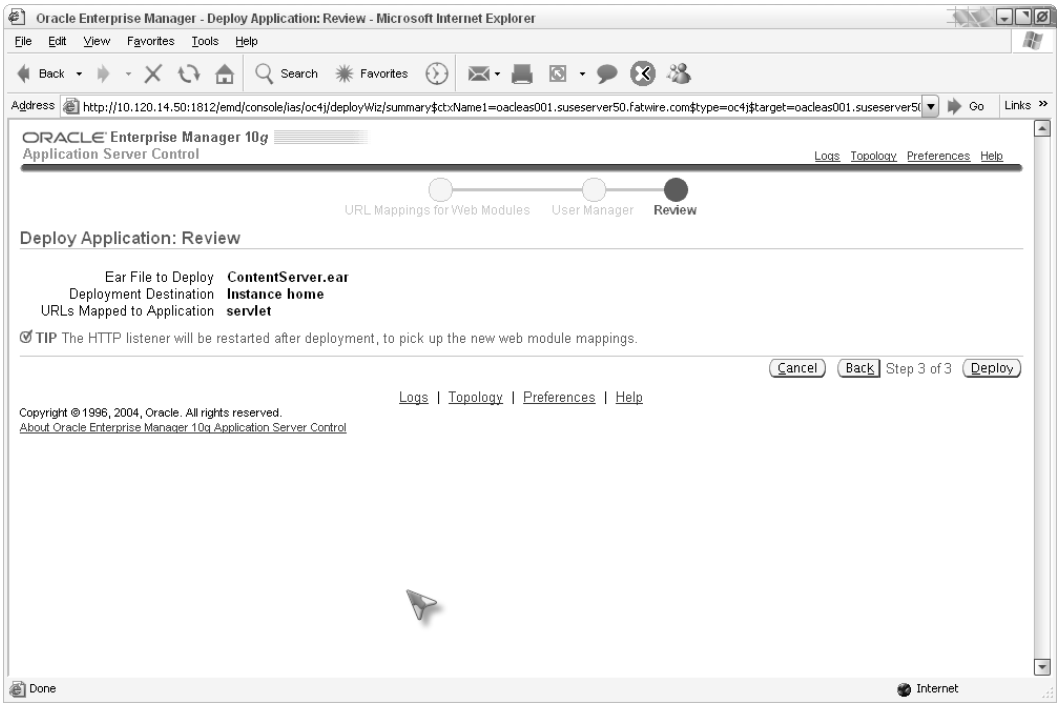
   b. Enter the name for this application and click **Continue**.



4. In "Context Mapping," enter the context root for this application and click **Finish**.



5. Review the options that you selected. If they are correct, click **Deploy**.



Chapter 6

Setting Up a Clustered Installation

This chapter shows you how to set up a clustered Content Server installation. The procedure entails creating a cluster instance. The instance is then migrated to other servers until the required number of cluster instances is obtained.

This chapter contains the following sections:

- Setting Up a Cluster Instance
- Migrating an Installation to Another Machine

Setting Up a Cluster Instance

This section shows you how to create a cluster instance that you can migrate to other servers (as shown in "Migrating an Installation to Another Machine," on page 55). The instance can be migrated to as many servers as necessary.

1. Determine whether the instance is attached to a farm by running:

   <ora home>/dcm/bin/dcmctl whichfarm

   a. If the instance is not attached to any existing farm, it returns "`Stand Alone`". Continue with step 2. Otherwise, continue with step b.

   b. If the instance does not return "`Stand Alone`", run:

      <ora home>/dcm/bin/dcmctl leaveFarm

2. Display the current farm id as a `<name>:<port>`

   <ora home>/dcm/bin/dcmctl getRepositoryID

   E.g.:

   The system returns: `suseserver50.fatwire.com:7101`

3. Join the farm that was returned in the previous step:

   <ora home>/dcm/bin/dcmctl joinFarm -r <name>:<port>

   Note

   If the join script fails, restart the OC4J process by using command above and try again with the integrated HTTP_Server stopped.

4. Confirm that the instance was added to the farm:

   <ora home>/dcm/bin/dcmctl whichFarm -v

   E.g.:

   ```
   Farm Name: .opt.software.App.Oracle.ora10a.dcm.repository
   Host Instance: oacleas001.suseserver50.fatwire.com
   Host Name: suseserver50.fatwire.com
   Repository Type: Distributed File Based (host)
   SSL In Use: false
   ```

5. Restart HTTP_Server if stopped in order to complete step 3.

6. Determine which clusters are defined:

   <ora home>/dcm/bin/dcmctl listclusters

7. Create a new cluster **with a unique name**:

   <ora home>/dcm/bin/dcmctl createCluster -cl <new clusters name>

8. List all available instances:

   <ora home>/dcm/bin/dcmctl listinstances

   E.g.:

   ```
   1
   Instance name: oacleas001.suseserver50.fatwire.com
   Cluster:
   Hostname:      suseserver50.fatwire.com
   Oracle Home:   /opt/software/App/Oracle/ora10a
   ```

9. Check that the instance is clusterable:

   <ora home>/dcm/bin/dcmctl isClusterable

   E.g.: `true`

10. Have an instance join the cluster:

    <ora home>/dcm/bin/dcmctl joinCluster -cl TestCluster -i <instance name>

11. List all instances in the cluster:

    <ora home>/dcm/bin/dcmctl listInstances -cl TestCluster

    E.g.:

    ```
    1
    Instance name: oacleas001.suseserver50.fatwire.com
    Cluster:       TestCluster
    Hostname:      suseserver50.fatwire.com
    Oracle Home:   /opt/software/App/Oracle/ora10a
    ```

12. Restart opmn now as a cluster:

    ./opmnctl startall

13. Check the status of the cluster by running:

    ./opmnctl @cluster status

14. Add any remaining instances that you wish to the OC4J cluster by running:

    <ora home>/dcm/bin/dcmctl createcomponent -ct oc4j -co <new clusters name> -i <instance name>

    E.g.:

    ```
    1
    Component Name: home2
    Component Type: OC4J
    Instance:       oacleas001.suseserver50.fatwire.com
    Cluster:        TestCluster
    ```

15. Check status of newly created instance:

    <ora home>/dcm/bin/dcmctl getstate -v

    E.g.:

    ```
    Current State for Instance:oacleas001.suseserver50.fatwire.com

    Component        Type          Up Status     In Sync Status
    ========================================================
    1 home2          OC4J          Down          True
    2 HTTP_Server    HTTP_Server   Up            True
    3 home           OC4J          Up            True
    ```

16. Check the status of the opmn components:

    ./opmnctl @cluster status

17. Start any new OC4J servers added in step 14:

    ./opmnctl startproc ias-component=OCJ4

18. Deploy `ContentServer.ear` to the newly created instance:

    <ora home>/dcm/bin/dcmctl deployapplication -f <cs ear location>/ContentServer.ear -a <deployment name> -co <new clusters name>

19. Duplicate any data sources found in the existing instance (normally named `home`) in the new instances. Do this by either:

    - Manually editing the `data-sources.xml` file (preferred). For instructions, see "XML-Based Procedure," on page 38.

      Note

      If you manually entered the data-sources, then restart opmn.

    - Using the graphical interface. For instructions, see "Web-Based Procedure," on page 39.

20. Edit the `default-web-site.xml` for each instance by replacing `ajp13` with `http`, as shown in the lines below:

    Original:

    ```
    <web-site port="7301" protocol="ajp13" display-name="OracleAS
      Java Web Site">
    ```

    New:

    ```
    <web-site port="7301" protocol="http" display-name="OracleAS
      Java Web Site">
    ```

Migrating an Installation to Another Machine

Related to clustering is moving Oracle AS (referred to in Oracle documentation as "migrating an installation from one machine to another"). What follows is a brief explanation how to move Oracle AS. For a more through explanation, refer to the Oracle AS documentation.

To move an installation

1. On the source machine:
   a. Remove the current instance from the farm:

      ```
      <ora home>/dcm/bin/dcmctl leavefarm
      ```

   b. Stop the emctl console.
   c. Stop all ipmnctl processes.
2. On the destination machine:
   a. Add `<ora home>/chgip/lib` to the library path. On Linux, add the path to `ld.so.conf`.
   b. Start ipmnctl dcm-daemon.
   c. Run **`<ora home>/chgip/chgiphost.sh -mid`** and enter the information that is requested at each prompt.
   d. Start the emctl console.
   e. Start all ipmnctl processes.
   f. Get the repository id:

      ```
      <ora home>/dcm/bin/dcmctl getRepositoryID
      ```

   g. Join the repository:

      ```
      <ora home>/dcm/bin/dcmctl joinfarm -r repository_ID
      ```

The instance has now been migrated to the new machine.

Part 3

Web Server

This part shows you how to install and configure your choice of supported HTTP servers: Oracle, IIS, and Sun ONE.

This part contains the following chapter:

- Chapter 7, “Installing and Configuring a Web Server”

Chapter 7

Installing and Configuring a Web Server

Oracle Application Server 10*g* comes with a copy of Oracle HTTP Server (OHS), IIS plugin, and Sun ONE plugin, all of which are supported web servers. However, the OHS is integrated (automatically installed) with Oracle Application Server 10*g*. However, there are many times when a remote web server is desired.

This chapter describes how to manually install and configure OHS (standalone instance), IIS, and Sun ONE HTTP servers.

This chapter contains the following sections:

- Installing and Configuring Oracle HTTP Server
- Configuring IIS Remote Plugin
- Configuring Sun ONE Remote Plugin

Note

Configuration instructions for OHS standalone also apply to the automatically installed (integrated) OHS. The instructions begin in section “B. Configuring OHS (Integrated and Standalone),” on page 61.

Installing and Configuring Oracle HTTP Server

Note

If you plan to use the automatically installed OHS, skip section "A. Installing OHS as a Standalone Instance," and continue with section "B. Configuring OHS (Integrated and Standalone)," on page 61.

A. Installing OHS as a Standalone Instance

This section contains an overview of how to install the OHS standalone instance. It does not cover each screen that is displayed during the process, only the main steps. If you have trouble following these instructions, refer to the Oracle AS documentation that comes with OHS Stand Alone.

1. Create a new user (in this example, `oracleas`).

   Unix creates a new user account named `oracleas` (you may choose any name that you wish, but in this guide we assume that you are using a user named `oracleas`).

2. Install OHS from the OracleAS Companion CD:
   - On Windows, run: **`setup.exe`**
   - On Unix,
     1) run: **`./runInstaller`**
     2) Create a new user named `oracle` and give this user permission to write to a location under which the `oraInventory` directory will be created.
3. Follow instructions in the Oracle product documentation to complete the following steps:
   a. Create an `oraInventory` directory.

   Note

   On Unix, run the script `orainstRoot.sh` in the `oraInventory` directory as root.

   b. Create an Oracle home directory.
4. Complete the installation.

   Note

   On Unix: Run the script `root.sh` as root.

B. Configuring OHS (Integrated and Standalone)

This section shows you how to configure OHS to communicate with the remote instance or with a cluster of Oracle Application Servers.

> **Note**
>
> Configuration instructions in this section apply to both the automatically and manually installed OHS.

1. Edit the file `ons.conf` located in `<ora home>/opmn/conf.`

   This file needs to contain a list of all hosts with which this server must communicate. The list must have the following format:

   ```
   nodes=<host_name | host_ip>[:port] [,<host_name |
          host_ip>[:port]] [, ...]
   ```

   Example: `nodes=10.120.14.50:6200,10.120.14.51:6200`

   > **Note**
   >
   > For information about which port is being used for `ajp13`, view the file `http-web-site.xml (in <ora home>/j2ee/<instance name>/config/http-web-site.xml)` on each Oracle AS server.

2. Edit the file `mod_oc4j.conf` located in `<ora home>/Apache/Apache/conf`, by adding a new `OC4jMount` line for each server in `ons.conf` with the context roots to be forwarded from the web server to the application server.

   Format: `Oc4jMount /<content root>/* <cluster or instance>://<host address>:<instance>`

   Example: `Oc4jMount /servlet/* instance://10.120.14.50:home`

3. Start the HTTP_SERVER:

   `<ora home>/opmn/bin/opmnctl startproc ias-component=HTTP_Server`

C. SSL (Optional)

> **Note**
>
> Instructions in this section apply to both the automatically and manually installed OHS.

1. Stop OHS if it is running:

   `<ora home>/opmn/bin/opmnctl stopproc ias-component=HTTP_Server`

2. Edit `opmn.xml` (located in `<ora home>/opmn/conf/`) by changing the entry `<ias-component id="HTTP_Server">` as follows:

   `Original:` **`<data id="start-mode" value="ssl-disabled"/>`**

```
New: <data id="start-mode" value="ssl-enabled"/>
```

Note

See Appendix A, "Oracle HTTP Server Self-Signed Certificates" for creating a self signed certificate for testing, or continue to the next section for a valid signed certificate.

3. Create an SSL Wallet and deploy it. If you need instructions, see section "D. Creating an SSL Wallet (Integrated and Standalone)."
4. Reload opmn configuration using: **`./opmnctl reload`**
5. Restart the HTTP_Server element:

   `./opmnctl startproc ias-component=HTTP_Server`

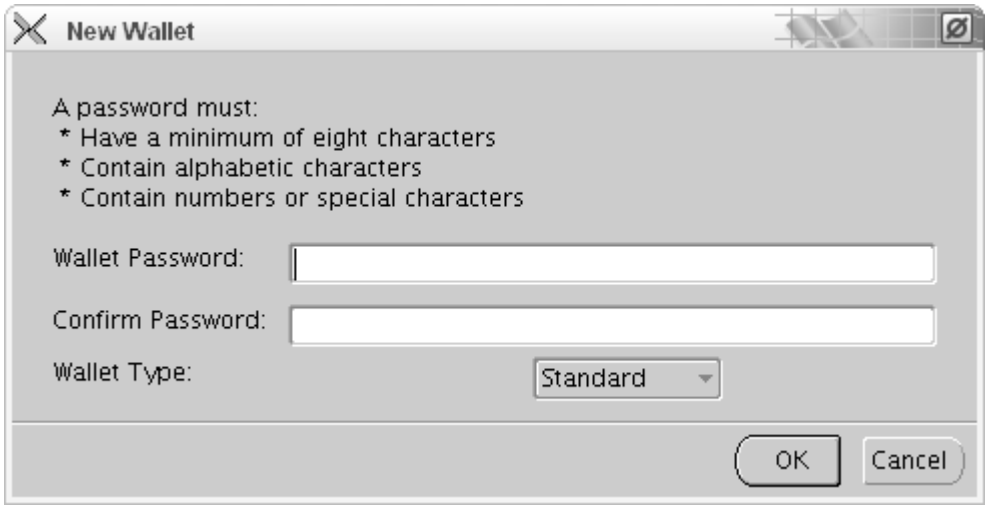
D. Creating an SSL Wallet (Integrated and Standalone)

Note

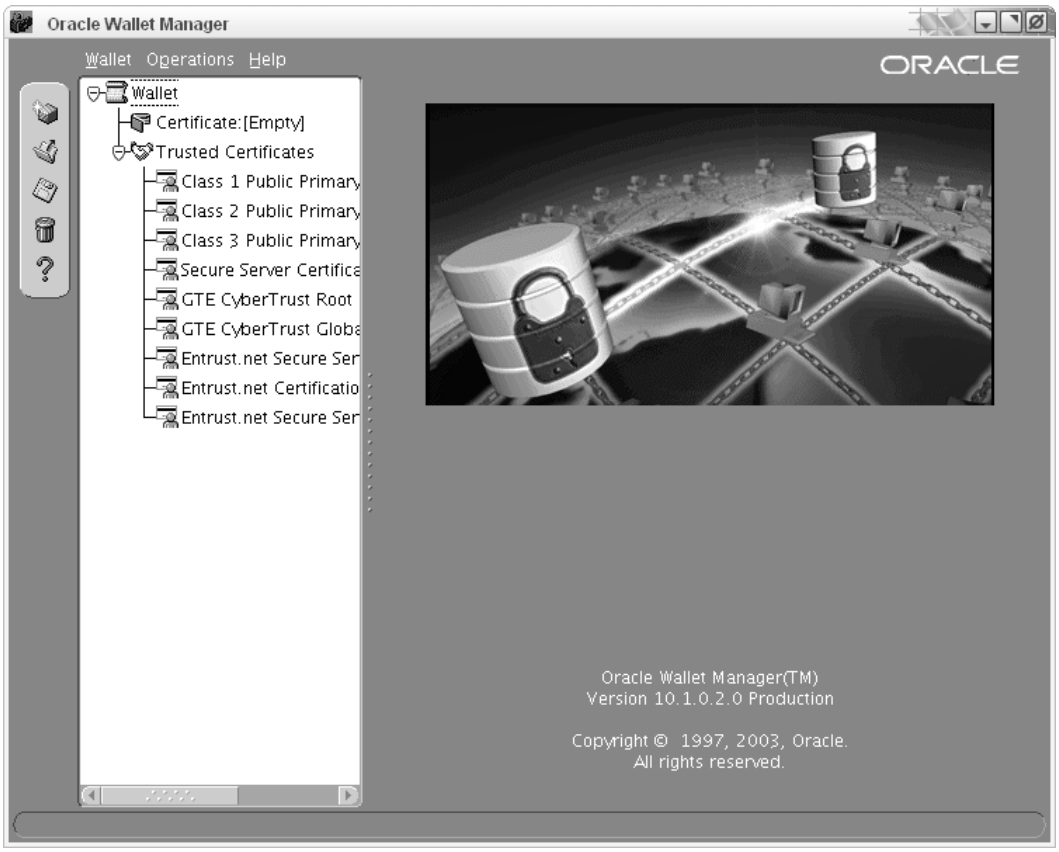
Instructions in this section apply to both the automatically and manually installed OHS.

This section steps you through the creation of a new wallet for OHS integrated or standalone. It assumes that you need to create a certificate request and will have that request signed by a trusted root authority. If you need to generate a self-signed certificate for installation or testing, refer to Appendix A, "Oracle HTTP Server Self-Signed Certificates."

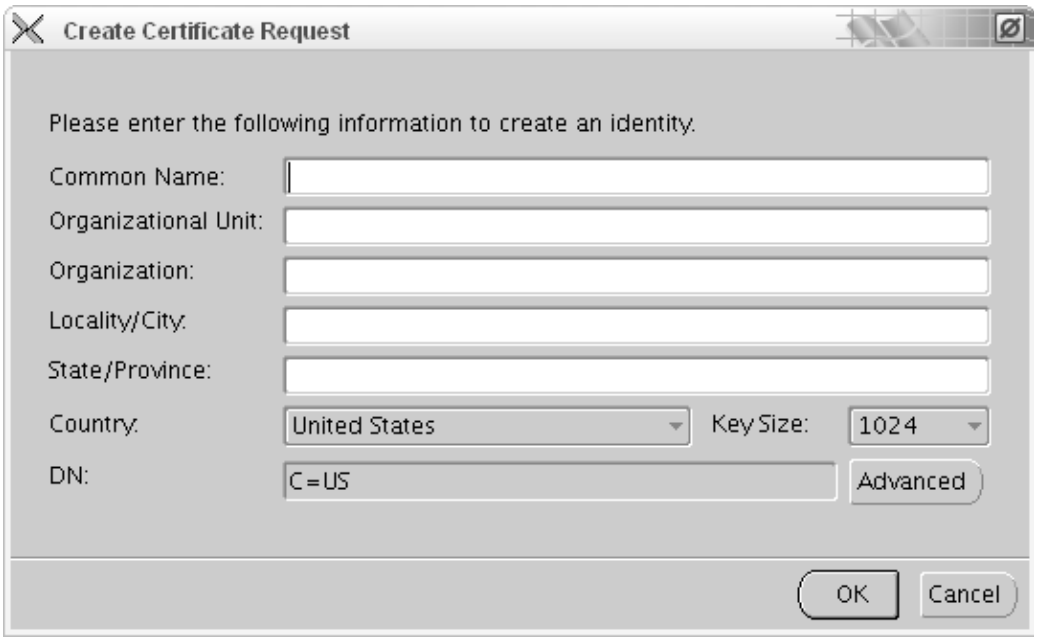
1. Start the **Oracle Wallet Manager: `<ora home>/bin/owm`**
2. Create a new wallet by clicking **Wallet** > **New**.
3. In the pop-up window enter a password, select type **Standard**, and click **OK**.



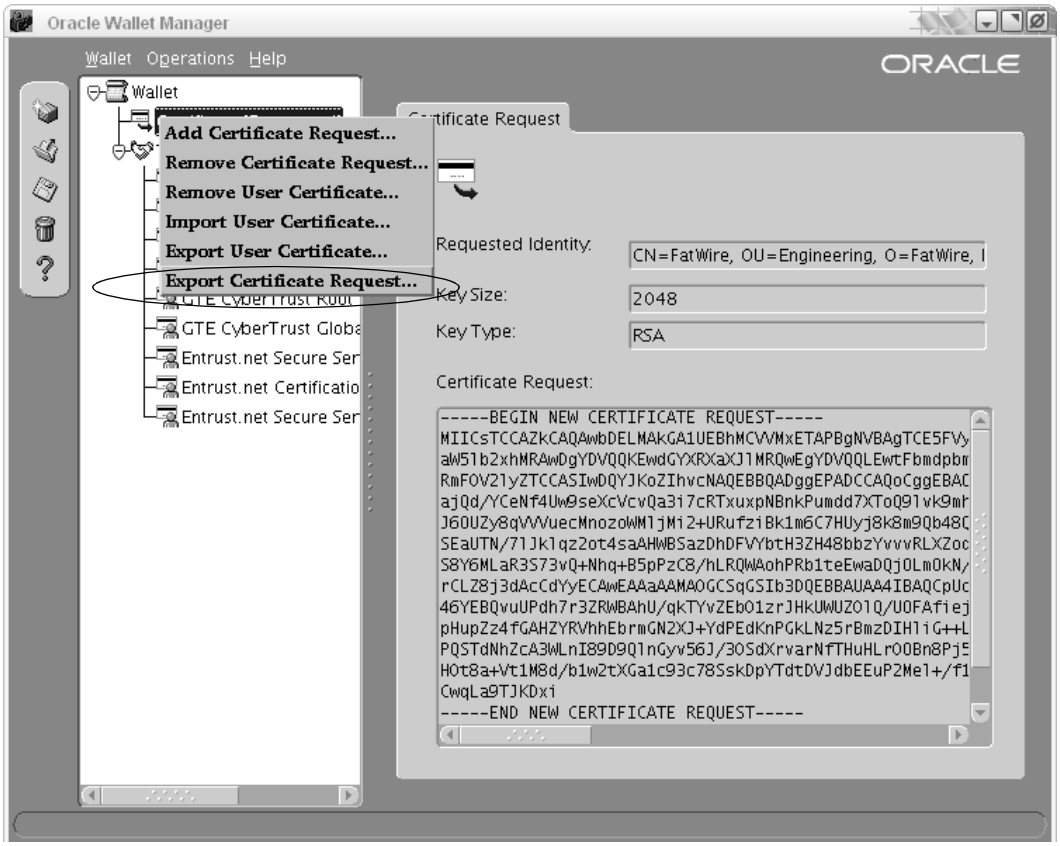
4. When asked if you wish to generate a certificate request, click **No**.
5. Back in the "Oracle Wallet Manager" window with the newly created wallet open, click **Wallet** on the menu bar **wallet** > **save as** and select the `<ora home>/Apache/Apache/conf/ssl.wlt` as the location to store the new file. You may use any name you desire for this wallet (in this document **mywallet** is assumed).
6. Generate a new certificate request by right-clicking on the tree item **Certificate** and selecting the item **Add Certificate Request …** .



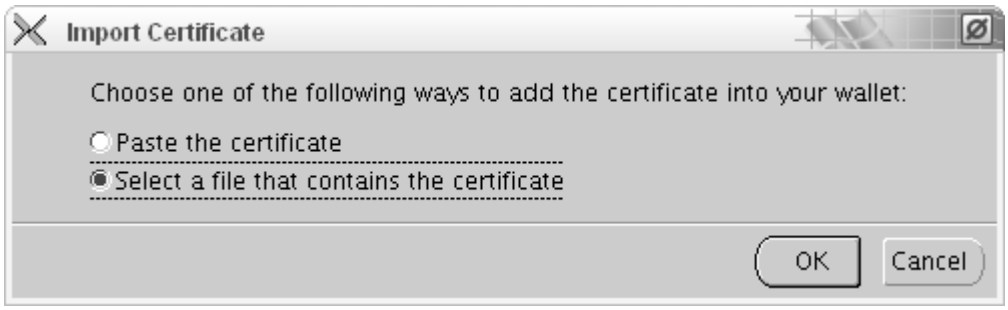
7. Fill in all fields in the resulting pop-up, then click **OK**.



8. Export the certificate request by right-clicking on the tree item **Certificate Request** > **Export Certificate Request…** You may choose any location and name that is convenient for you. You will have to submit this request to your chosen root authority.



9. Once you have the received the signed user certificate from your root authority, right-click the tree item **Certificate Request** and choose **Select a file that contains the certificate**. Browse to the location of the signed user certificate and click **OK**.



10. Save your newly verified certificate (on the menu bar click **Wallet** > **Save**).
11. Following the steps below, modify the OHS to find the wallet that was created in steps 1–10.
    a. Open the file `<ora home>/Apache/Apache/conf/ssl.conf` in an editor.
    b. Locate the line: `SSLWallet file:` (default location is around line 171) and change it so that the default is the name of the wallet created above. In this example the name of the new wallet is `mywallet`. For example:

       Original:
       ```
       SSLWallet file:/opt/software/App/Oracle/ora10a/Apache/
       Apache/conf/ssl.wlt/default
       ```

       Modified:
       ```
       SSLWallet file:/opt/software/App/Oracle/ora10a/Apache/
       Apache/conf/ssl.wlt/myWallet
       ```

    c. Save changes to `<ora home>/Apache/Apache/conf/ssl.conf`
12. Restart the OHS instance:

    `<ora home>/bin/opmnctl startproc ias-component=HTTP_Server`

Configuring IIS Remote Plugin

This section shows you how to configure IIS as a front end for an Oracle AS server.

1. Download the plugin (which is also located on the Oracle Application Server 10*g* Companion CD) and copy the plugin to a location on the local file system.
2. Create a new file `opii.conf` that contains information about which context roots to forward and where. Each context root needs to be on its own line.

   Format: **`Oc4jMount /<content root>/* ajp13://<host name>:<port>`**
   Example: **`Oc4jMount /j2ee/* ajp13://localhost:3000`**

   Note

   For information about which port is being used for `ajp13`, view the file `http-web-site.xml` (in `<ora home>/j2ee/<instance name>/config/http-web-site.xml`) on each Oracle AS server.

3. Open `regedit` and add the following hierarchy:

   `HKEY_LOCAL_MACHINE\SOFTWARE\Oracle\OPII`

   a. Create a new string value pair: `server_defs:<location of conf>\opii.conf`
   b. Create a new string value pair: `log_file:<location of log>\opii.log`
   c. Optional: create a new string value pair: `log_level:(debug or error)`
   d. Optional: Debugging the status page: Create a new string value pair: `status_uri:/oc4j-service`
4. Add a new filter to the IIS instance named `opii.dll`. As a value enter `<path of opii>\opii.dll`.

   Note

   For IIS6, all the dlls in `<oracle home>\bin` must be executable by the user “NETWORK SERVICE.”

5. Restart the entire IIS Service (not just the program instances).

Configuring Sun ONE Remote Plugin

This section shows you how to configure Sun ONE Web Server as a front end for an Oracle AS.

1. Locate the plugin on the Oracle Application Server 10*g* Companion CD:
   - Unix systems: `/plugins/solaris/sunone`
   - Windows: `/plugins/win32/sunone`
2. Copy the plugin (`Solaris: opii.so`; Windows: `opii_sunone.dll`) to a location on the local file system. The location such as `<sunone home>\plugin` must be readable by the Sun ONE listener (create the directory if needed).
3. Edit the file `magnus.conf` for the instance which is to use the plugin by adding the following lines. Make sure to replace `<instance name>` and `<SunOne home>` with the appropriate values for your system:

```
Init fn="load-modules" shlib="/<sunone home>/plugins/opii.so"
 funcs=opii_init,opii_objecttype,opii_service,opii_
 child_init

Init fn="opii_init" log_file="/<sunone home>/<instance name>/
 logs/opii.log" log_level=error server_defs="/<sunone
 home>/<instance name>/config/opii.conf" Init
 fn="opii_child_init" LateInit=yes
```

4. Edit the file `obj.conf`, for the instance which is to use the plugin by adding the following lines as explained below:
   a. Before the first line that begins with `ObjectType`, add the following:
   ```
   ObjectType fn=opii_objecttype
   ```
   b. Before the first line that begins with Service, add the following:
   ```
   Service type="oracle/opii" fn="opii_service"
   UseOutStreamSize=8192
   ```
   c. Optional for debugging the status page:
   Before the first list that begins with `NameTrans`, add the following:
   ```
   NameTrans fn=assign-name from="/oc4j-service" name=
    "opii-status"
   ```
   d. To the end of the file, add:
   ```
   <Object name="opii-status">

   Service fn="opii_status_service"
     </Object>
   ```
5. Restart the Sun ONE instance.

Part 4

Content Server

This part shows you how to proceed through the installation of Content Server. It contains the following chapter:

- Chapter 8, “Installing Content Server”

Chapter 8

Installing Content Server

This chapter steps you through the installation of Content Server on Oracle Application Server 10*g*.

This chapter contains the following sections:

- Step I. Complete Pre-Installation Procedures
- Step II. Install Content Server
- Step III. Complete Post-Installation Procedures

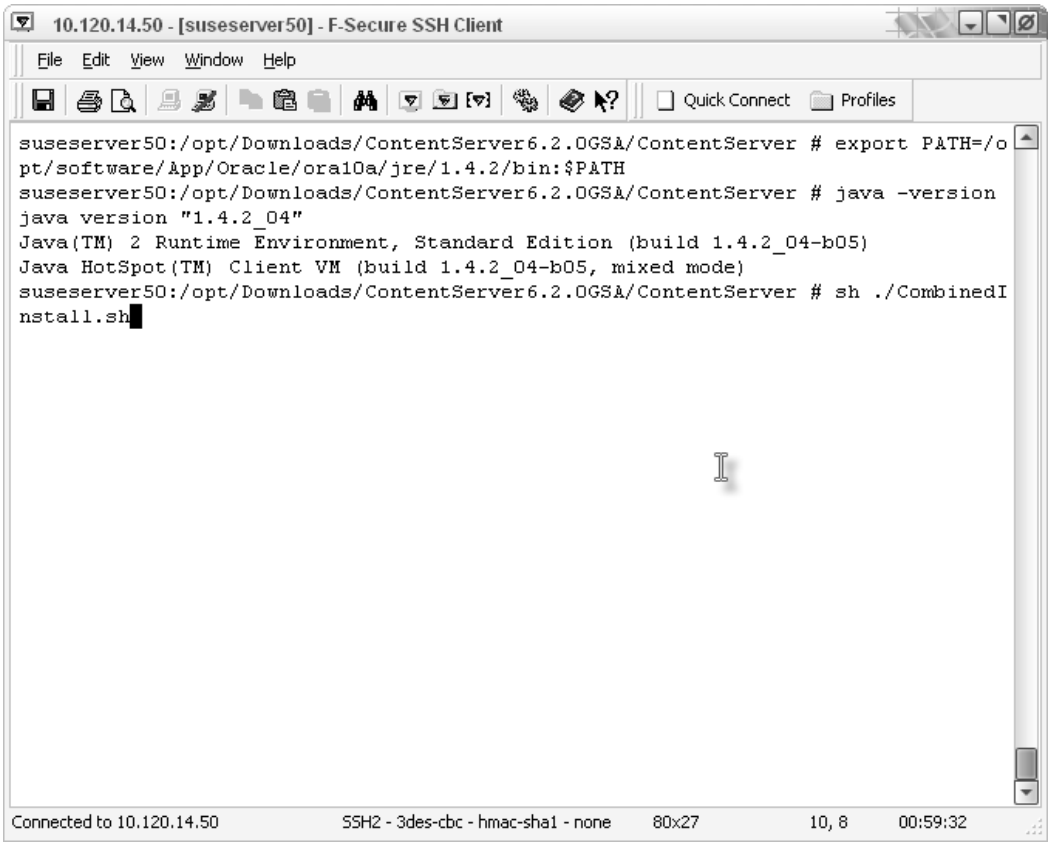
Step I. Complete Pre-Installation Procedures

Before installing Content Server, make sure you have completed the following:

- Performed steps 1–5 in the section "Installation Steps," on page 7.
- Obtained a license with the correct IP address and ports
- Created a valid directory into which you plan to install Content Server
- For clustered installations: You have created a valid shared directory into which you plan to install the Content Server shared file system.

Step II. Install Content Server

1. Start the Content Server installer by changing to the Content Server installer directory running **`./CombinedInstall.sh`** on Solaris (**`CombinedInstall.bat`** on Windows).



2. Select the option **Install FatWire Products** and click **Next**.

3. Choose the directory where you would like Content Server to be installed. This is the directory that was created in the pre-installation step on page 72. Click **Next**.

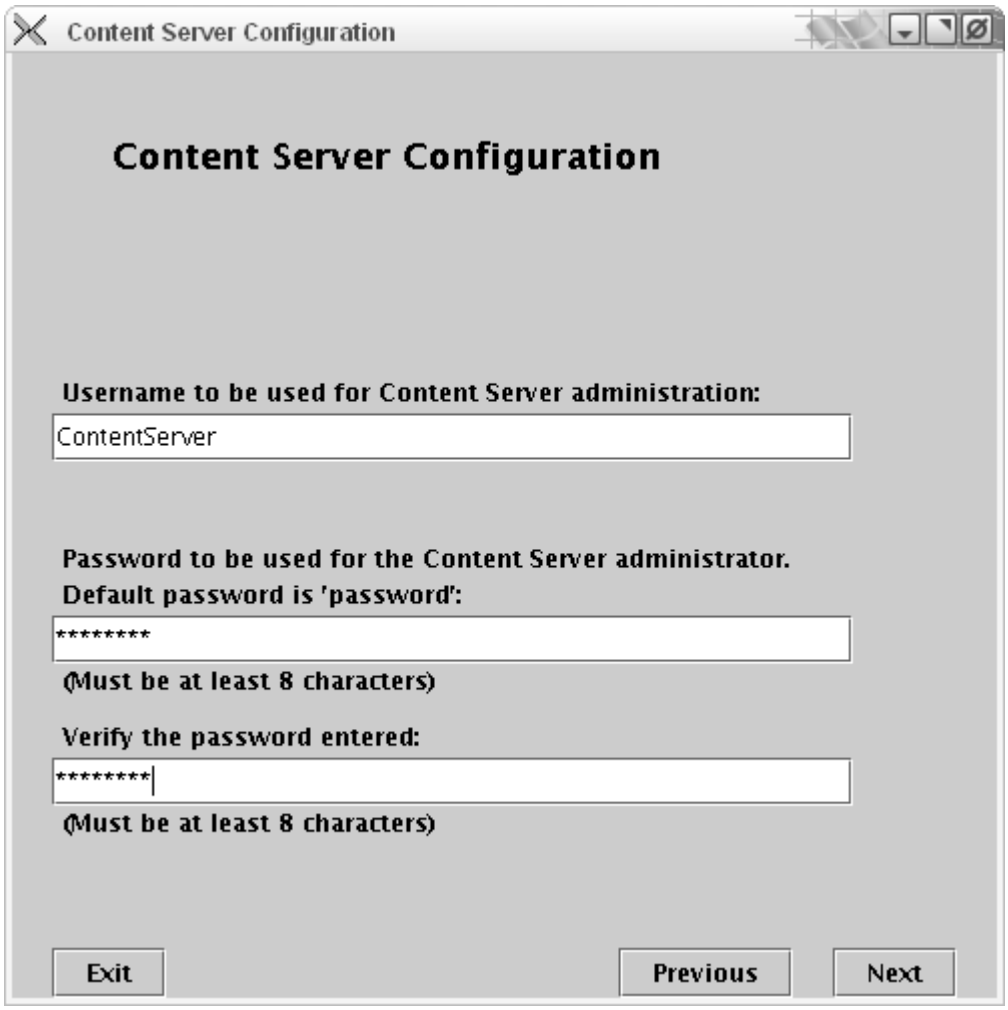
4. Enter the location of the `FWLicense.xml` file which you received from FatWire and click **Next**.

5. Select the products to install and click **Next**.

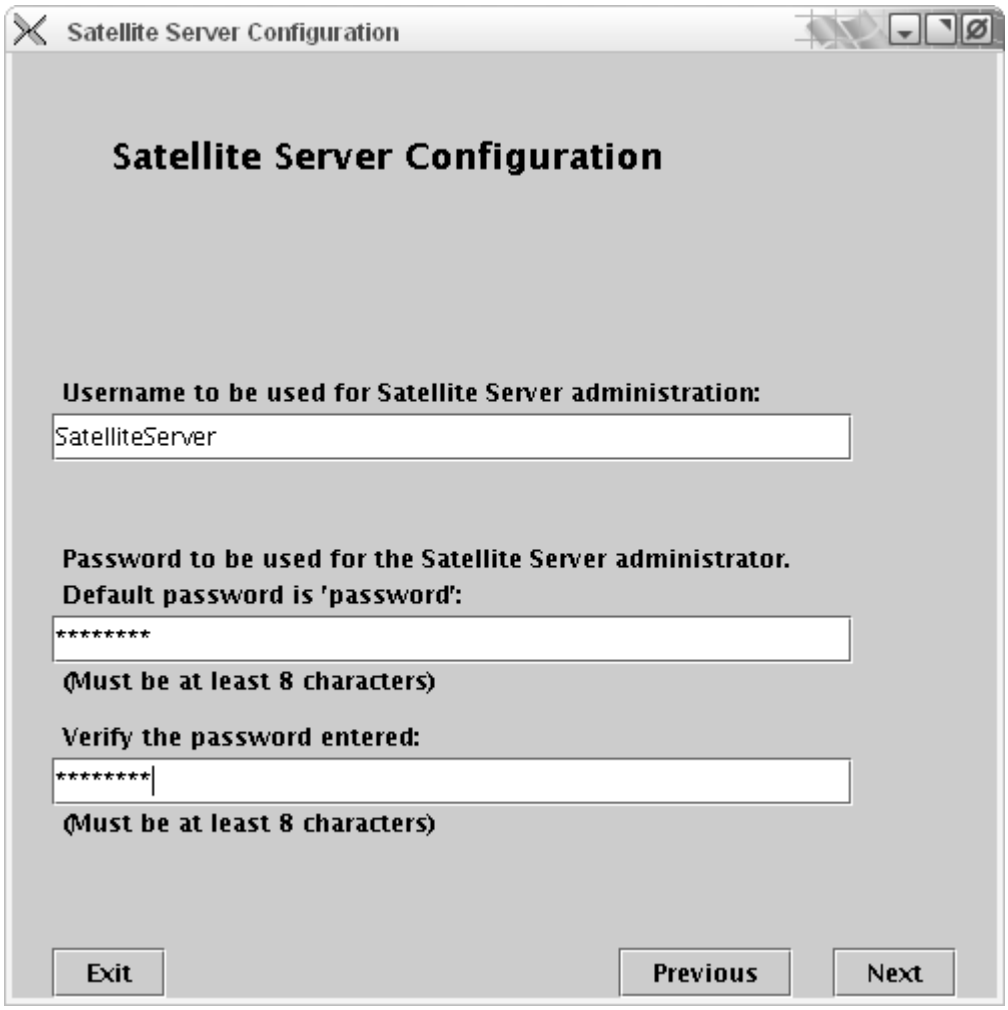
6. Select the type of installation. If this is a non-clustered installation or the first machine in a cluster, select **Single Server**. Otherwise select **Cluster Member**. Click **Next**.

7. Select any special installation options (it is normally safe to leave the defaults) and click **Next**.

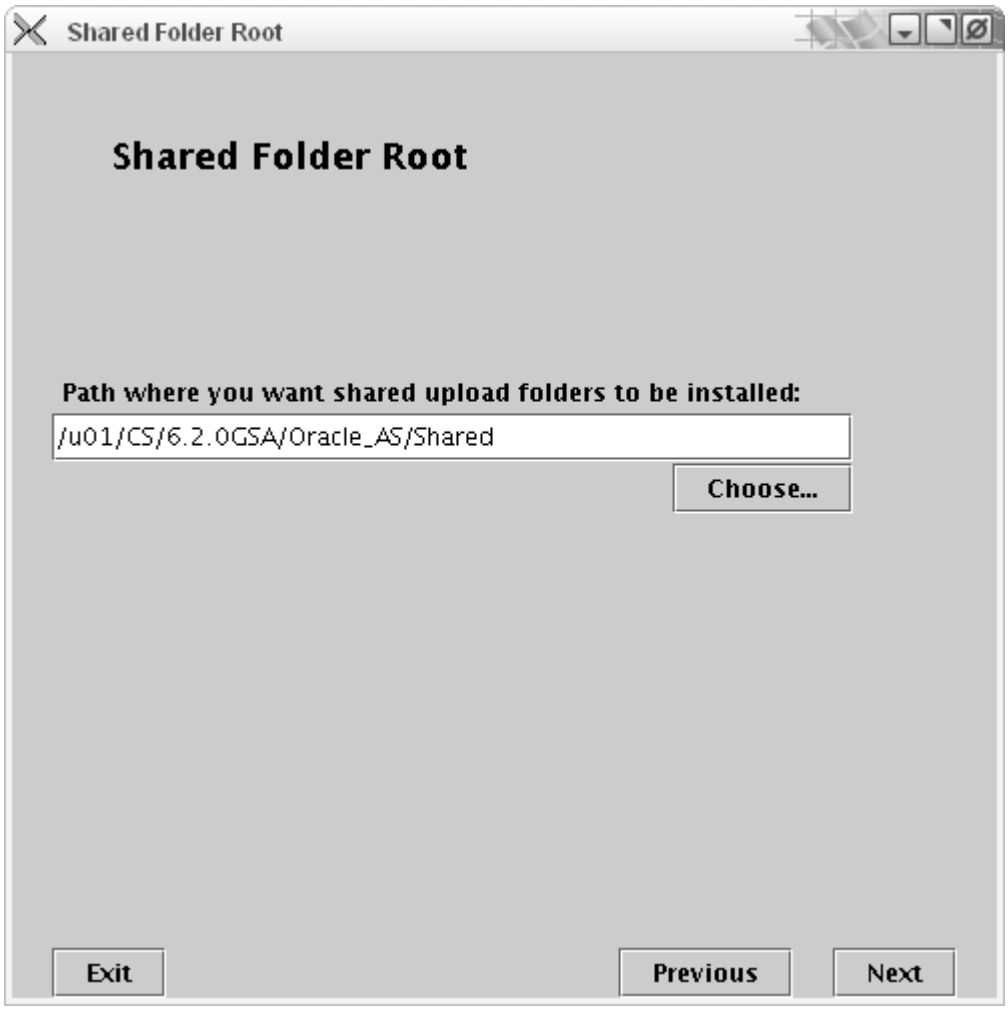
8. Enter a password for the Content Server user. The default value of the password field is `password`, but should be changed for security reasons. Click **Next**.



9. Enter a password for the Satellite Server user. The default value of the password field is `password`, but should be changed for security reasons. Click **Next**.



10. Enter the location of the shared file system that was created in the pre-installation steps on page 72, and click **Next**.



11. Inform Content Server as to how and where it will be installed (this means entering the DNS name of the host on which you will access it and the port).

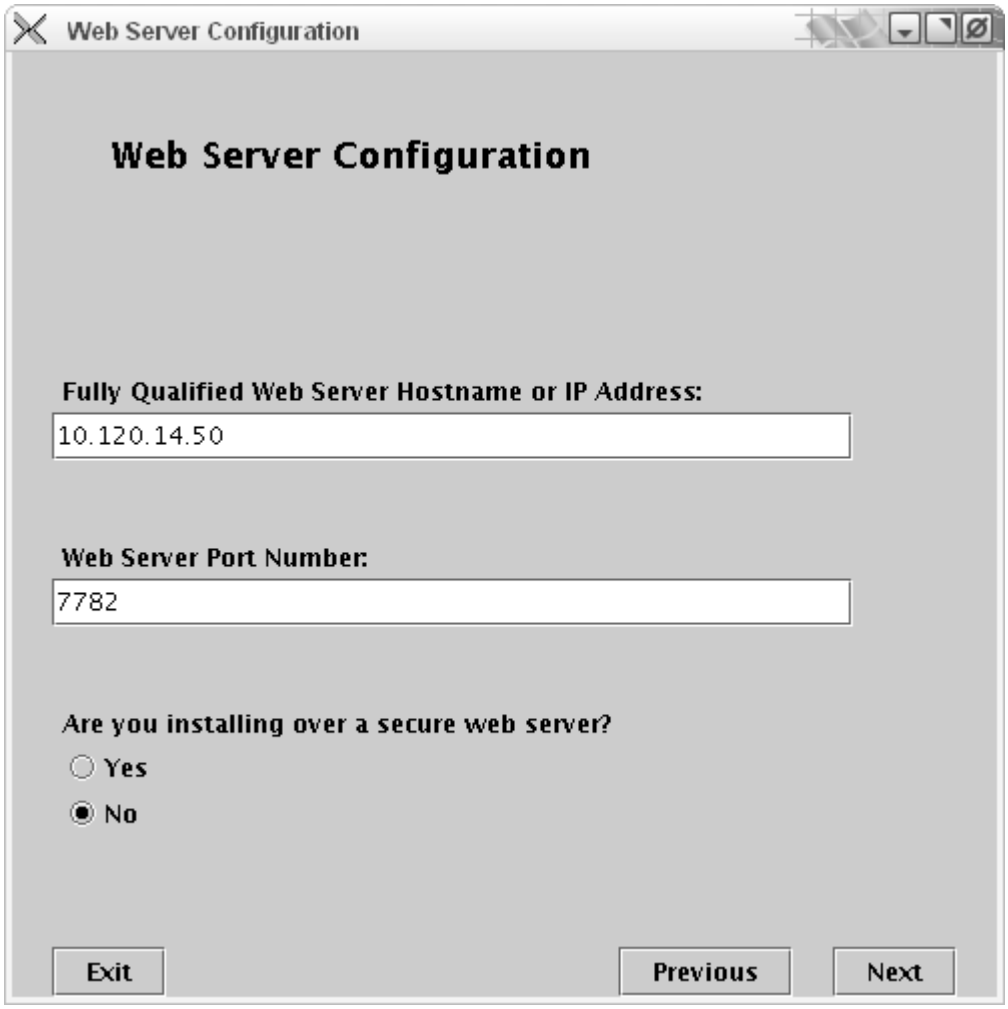
> **Note**
>
> For installation, it is normally a good idea to use the application server directly and bypass the remote web server, as this will eliminate third party connectivity issues that may affect the installation.

a. Enter the DNS Name.

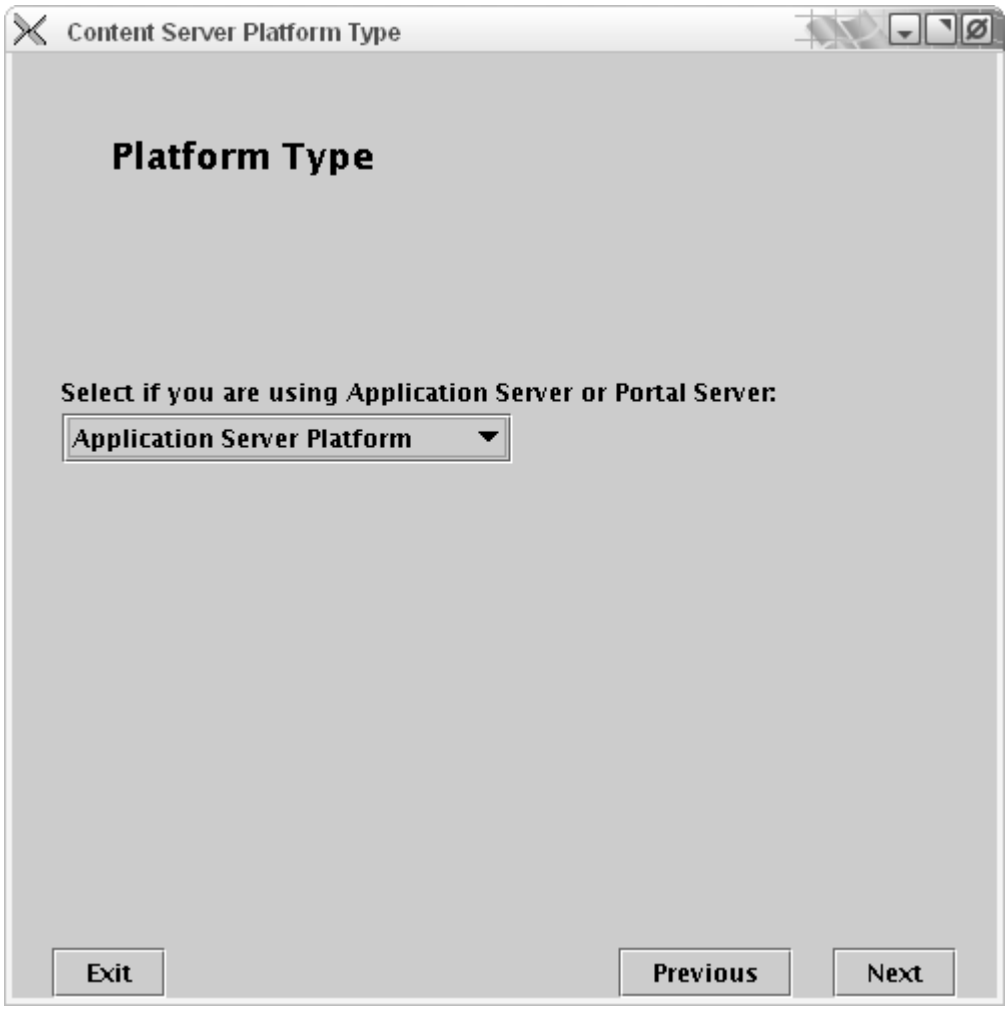
b. Enter the web server port number. The port on which Oracle AS is running was displayed at the end of the Oracle AS installation and can be found by inspecting the file `<ora home>/install/portlist.ini` on the application server.

c. If you are installing over a secure web server, select **Yes**.

d. Click **Next**.

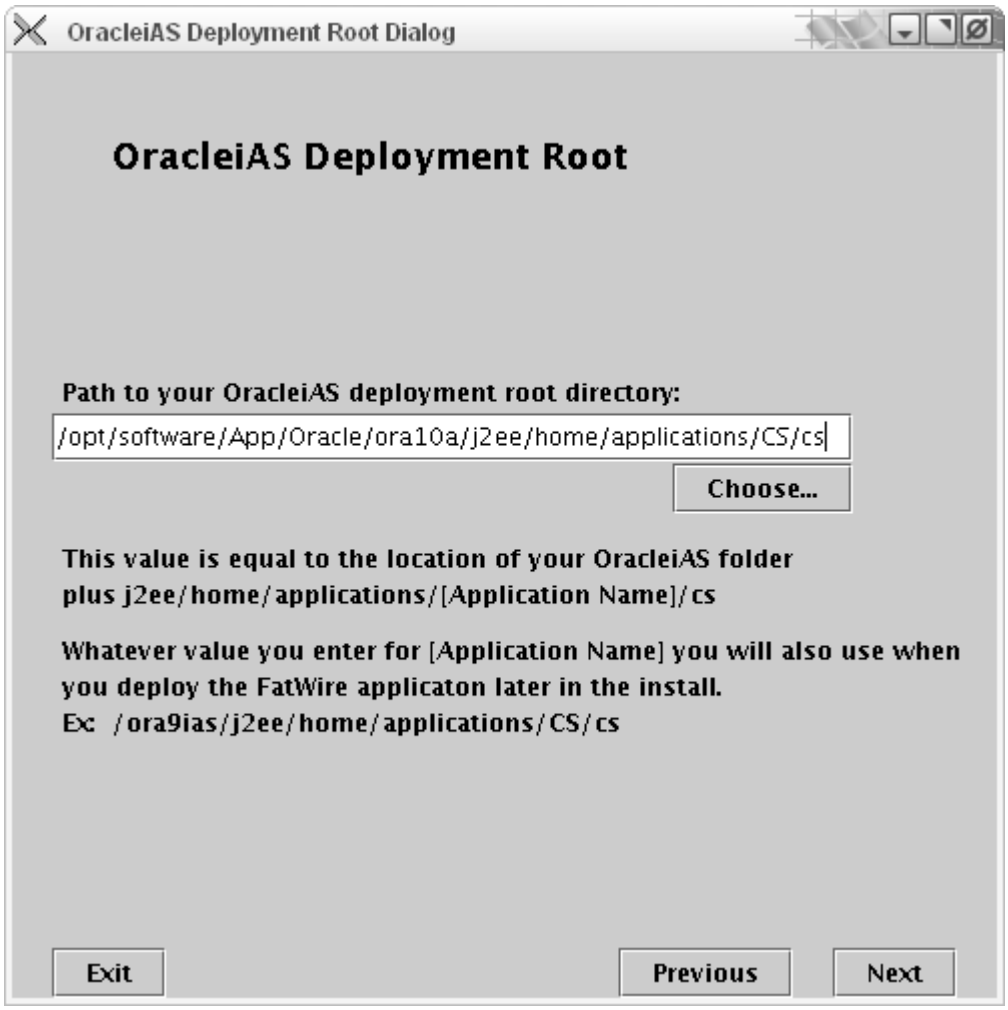


12. Leave the default **Application Server Platform** selected and click **Next**.

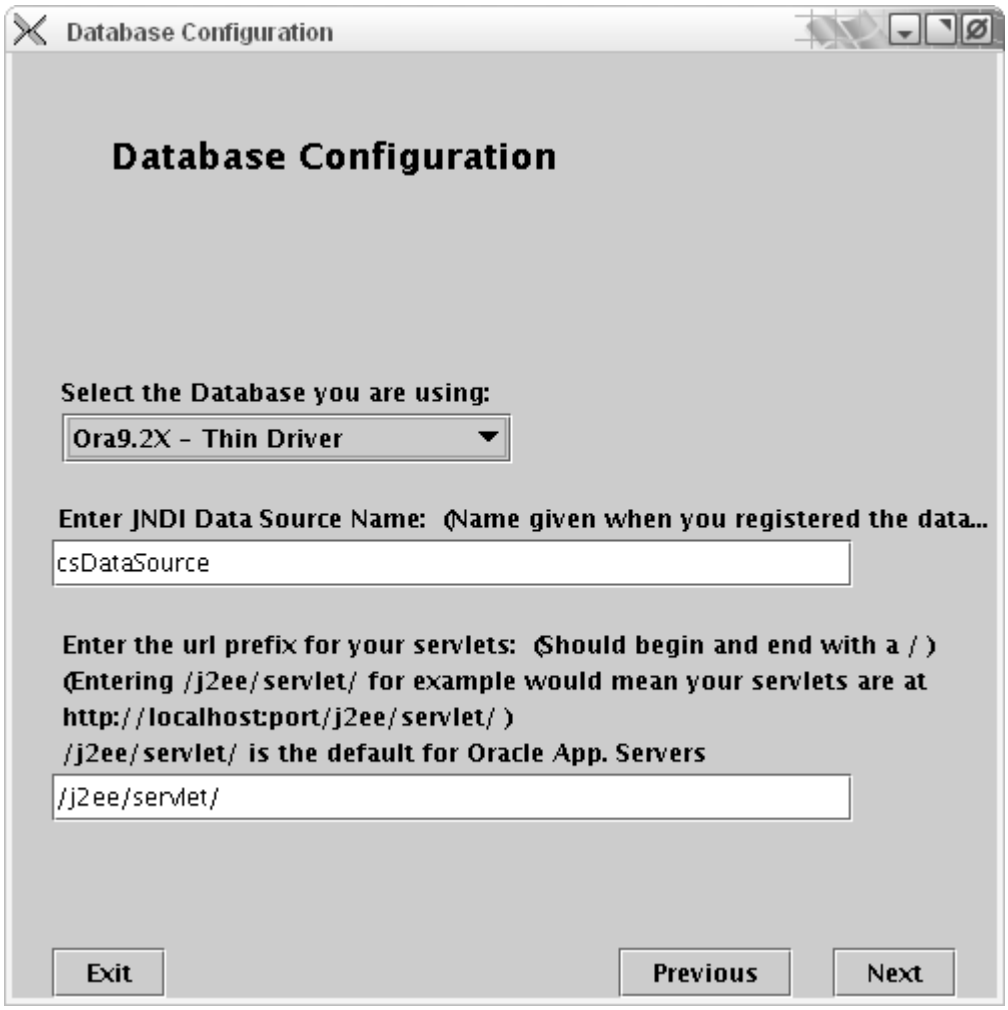


13. Select **Oracle 10*g* Application Server** from the pull-down menu and click **Next**.

14. Enter the application deployment root directory (this will always be: `<ora home>/j2ee/<instance name>/applications/<application name>/CS`). Click **Next**.



15. Enter the name of your data source and the context root for this product. The default context root in Oracle is `/j2ee/servlet/`. However, FatWire suggests that you change this to `/cs`. Click **Next**.



16. Choose the components according to the products that you purchased and click **Next**.

17. Select the type of installation this will be. Do one of the following:

- Select **Content Management** if you are installing Content Server on either a development or management (staging) system *and* you wish to install sample sites and their assets on the system. (By selecting the option, you allow sample sites and assets to be installed later in the installation process.) Complete the following steps:
  - **1)** Click **Next**.
  - **2)** In the window that opens (not shown), follow instructions to choose the samples you want to have installed.
  - **3)** When the "Content Server Applications Install" window opens, go to step 18.
- Deselect **Content Management** if you are installing Content Server on a delivery (production) system, or any system where sample sites and assets are unnecessary. (By deselecting the option, you prevent sample sites and assets from being installed.) Click **Next**.

18. You are now ready to begin the installation:

a. Click **Install**

Note

When the following pop-up window appears (halfway through the installation), it means that the Content Server base has been installed. You now need to deploy the application and test that it can connect to the database. Continue with step b.

b. Deploy the `ContentServer.ear` file located in `<your Content Server installation directory>/ominstall/apps/`. For instructions, see Chapter 5, "Deploying Applications."

c. In order for the installation to succeed, `libFTfilelock.so` must be in the `LD_LIBRARY_PATH`. To place the file, do the following:

1) Locate the file in the directory from which you started the Content Server installation.

2) Go to `bin/<machine arch>/` and copy the file `libFTfilelock.so` to `<oracle appserver installation directory>/lib`.

3) Log in to the Oracle Appserver Console and navigate to the OC4J instance on which you installed Content Server (default is home).

4) Select the **Administration** tab followed by the option **Server Properties**.

5) In the section "Environment Properties," add two new sections:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH: <oracle appserver
   installation directory>/lib
```

```
PATH=$PATH:<oracle appserver installation directory>/lib
```

6) Click the **Apply** button.

7) Restart the instance when prompted to do so.

d. Once the application is deployed, test that it works and can connect to the database. To test the application, do the following:

- Go to `http://<DNS name>:<port>/<context root>/HelloCS`

  This tests whether Content Server deployed successfully.

- Fill in the values with those used during the installation above:

```
http://<DNS name>:<port>/<context root>/
  Cataloganager?ftcmd=pingdb
```

  This tests whether Content Server can communicate with the database.

- If both tests work correctly, you are ready to continue the installation.

e. When the final pop-up is displayed stating that the installation was successful, click **OK** to close the pop-up window. Click **Exit** to close the Content Server installation window.

Step III. Complete Post-Installation Procedures

1. Verify the installation by logging in to Content Server as the administrator:

   `http://<hostname>:<port>/<context root>Xcelerate/LoginPage.html`

   Login name: **fwadmin**

   Password: **xceladmin**

2. If you plan to use the Verity search engine, following installation guidelines in Appendix B, “Installing Verity Search Engine.”

Appendixes

This section contains the following appendixes:

- Appendix A, “Oracle HTTP Server Self-Signed Certificates”
- Appendix B, “Installing Verity Search Engine”

Appendix A

Oracle HTTP Server Self-Signed Certificates

This appendix shows you how to create a self-signed certificate for Oracle OHS servers. It uses a combination of the Oracle Wallet command line and graphical interfaces as well as OpenSSL (an open source that can be downloaded for a number of platforms).

1. Create a wallet for OHS by invoking the **orapki** command as follows and replacing <ora home> and <name> with the correct values for your configuration:

```
# <ora home>/bin/orapki wallet create  \
-wallet <ora home>/Apache/Apache/conf/ssl.wlt/<name> -
 auto_login
```

2. Generate a trusted root authority, change to a temporary location and execute the following commands in the following order:

```
# mkdir demoCA
# cd demoCA/
# mkdir certs
# mkdir crl
# touch index.txt
# mkdir newcerts
# echo "01" > serial
# mkdir private
# cd ..
# openssl genrsa -out ca.key 1024
# openssl req -new -x509 -key ca.key -out cacert.pem
```

3. You will be asked to enter information that will be incorporated into your certificate request. The information that you will enter is what is called a “Distinguished Name” or a DN. There are a number of fields, but you can leave some blank; others will have a default value. Press **Enter** if you wish to use the defaults. The following fields must be filled in:

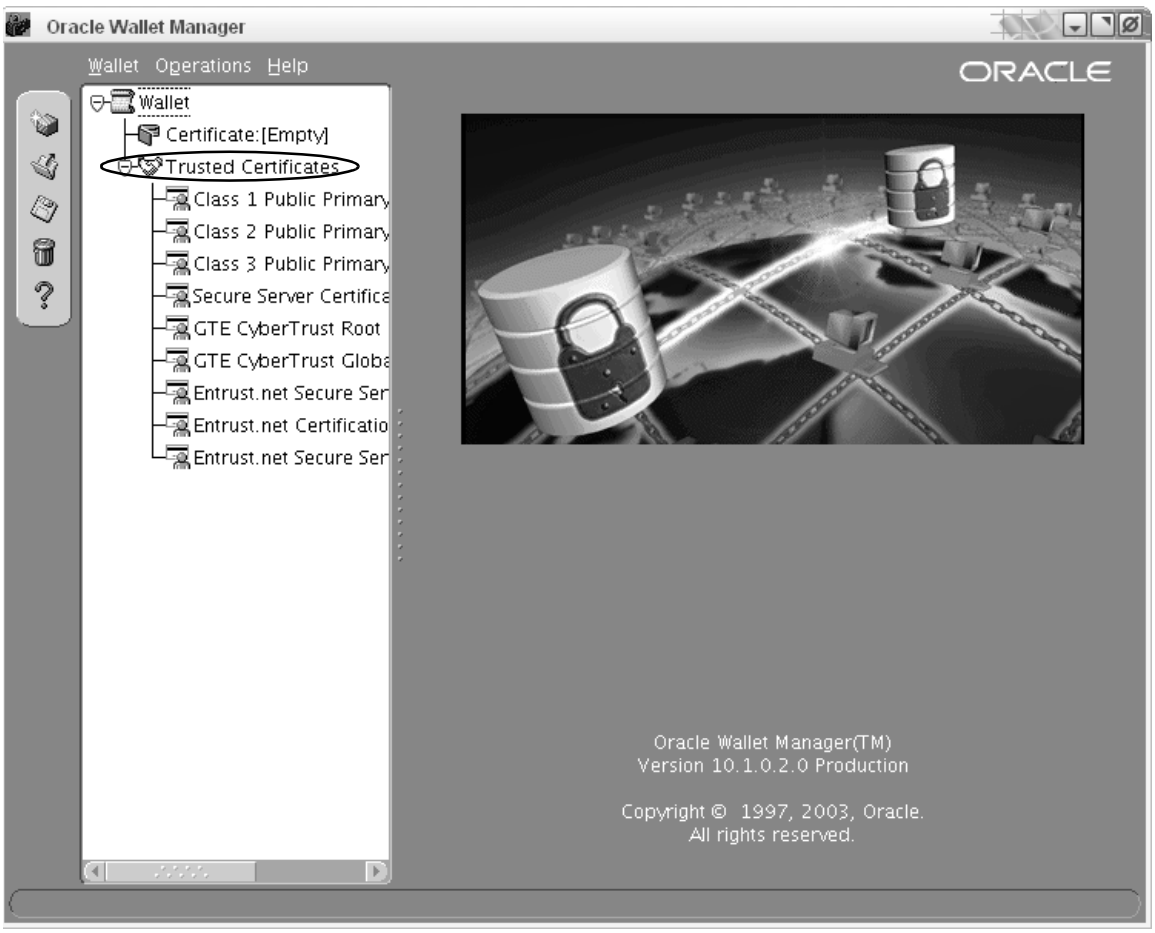
```
Country Name (2 letter code) [ ]:
State or Province Name (full name) [ ]:
Locality Name (e.g., city) [ ]:
Organization Name (e.g., company) [ ]:
```

```
Organizational Unit Name (e.g., section) [ ]:
Common Name (e.g., YOUR name) [ ]:
Email Address [ ]:
```

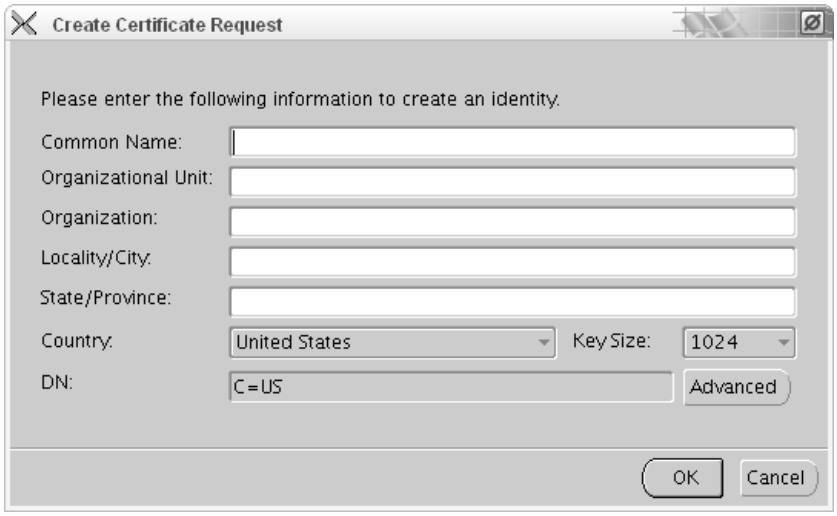
4. Start the Oracle Wallet Manager product:

   `<ora home>/bin/owm`

5. Save the new wallet to the disk in the `<ora home>/Apache/Apache/conf/ssl.wlt/` directory. Click **Wallet** > **Save As**, browse to the location above, and save the wallet with any unique name.
6. Import `demoCA/cacert.pem` into the OHS wallet as a trusted root (right-click the **Trusted Certificates** header on the tree, and click **Import Trusted Root…** from the drop-down menu).



7. Generate a Request Certificate, right-click **Certificate** in the tree view, and click **Add Certificate Request**.
8. In the "Create Certificate Request" window, fill in the fields. Make sure that the field values exactly match those in step 3.



9. Sign the exported Certificate Request, using the **openssh** command line utility. This will prompt you for a number of fields. Make sure that the values you enter exactly match those in step 3.

   openssl ca -in myrequest -out certified_myrequest.pem -keyfile ca.key

10. Using configuration from `/etc/ssl/openssl.cnf`, check that the request matches the signature.

```
Signature ok
The Subjects Distinguished Name is as follows
countryName           :PRINTABLE:'Country '
stateOrProvinceName   :PRINTABLE:'State'
localityName          :PRINTABLE:'Locality '
organizationName      :PRINTABLE:'Organization '
organizationalUnitName:PRINTABLE:'Department'
commonName            :PRINTABLE:'name'
Certificate is to be certified until Apr 10 11:58:51 2008
    GMT (1095 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

11. Edit the file `certified_myrequest.pem` by removing everything except `Begin Certificate`, `End Certificate`, and the text in between; that is, keep the following:

```
-----BEGIN CERTIFICATE---
<text>
-----END CERTIFICATE-----
```

12. Import the signed certificate (`certified_myrequest`), right-click **Certificate** in the tree view, and click **Import User Certificate…**
13. Save the wallet (from the top menu bar, select **Wallet > Save**).
14. Configure the Oracle HTTP Server with SSL. For instructions see, “SSL (Optional),” on page 61 and “Creating an SSL Wallet (Integrated and Standalone),” on page 62.

Appendix B

Installing Verity Search Engine

The Verity Search Engine comes with a set of installation notes; however, a few changes and clarifications need to be made for it to work properly with Oracle Application Server. The instructions below for installing the Verity Search Engine supplement the instructions in the Verity product documentation.

1. Copy the `libFTVeritySearch.so` file into the `<ora home>/lib` directory.
2. Copy the jar file `Verityse.jar` into the third-party jar location `<ora home>/j2ee/<instance name>/applib/`
3. Edit the file `<ora home>/opmn/bin/opmnctl` as follows:

   After the section dealing with setting `LD_LIBRARY_PATH`, add the following two lines:

   ```
   # verity Additions
   CSVERITYPATH=<content server installation directory>/VerityK2/
     <_platform>
   LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$CSVERITYPATH/
     filters:$CSVERITYPATH/bin
   ```