Content Server / Spark

Version: 6.3

Configuration Guide: Third-Party Software

Document Revision Date: Jan. 31, 2006

FatWire SOFTWARE

Configuration Guide: Third-Party Software
Document Revision Date: Jan. 31, 2006
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 2. Installing a Web Server

Part 3. LDAP Integration

About This Guide

This guide contains information about installing and configuring third-party software specifically for use by Content Server and Spark. Instructions in this guide supplement the instructions in the Content Server and Spark installation guides. The steps you will follow show you how to create and configure the supported databases, install supported web servers, and integrate Content Server with LDAP servers.

Who Should Use This Guide

This guide is for installation engineers who have experience installing and configuring enterprise-level software, including databases, database drivers, application servers, and portal servers, and LDAP servers.

Graphics Used in This Guide

Graphics in this guide are screen captures of dialog boxes and similar windows that you will interact with during the installation or configuration process. These graphics are presented to help you follow the installation and configuration processes. They are not intended to be sources of information such as parameter values, options to select, and product version numbers.

Technical Support

Help is available from FatWire Technical Support at the following website:

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

Part 1

Creating and Configuring a Database

Content Server and Spark each require access to a supported database that is specifically configured for the product. Supported databases include:

- Oracle 9
- Oracle 10*g*
- Microsoft SQL Server 2000 SP3 and SP4
- DB2
- My SQL Anywhere (Spark only)

The databases listed above are not configured for production, but are set up with full permissions. In practice, the permissions can be curtailed for the user that Content Server (or Spark) will use to access a database. However, the following rights must exist: ability to create, modify, and delete tables and indexes.

If you need instructions on installing a supported database, refer to the product documentation.

Instructions on creating and configuring the databases for Content Server (or Spark) are given in the chapters of this guide. Because database configuration is identical across different application servers, refer to the correct chapter to set up the database of your choice.

This guide contains the following chapters:

- Chapter 1, “Creating and Configuring an Oracle 9 Database”
- Chapter 2, “Creating and Configuring an Oracle 10g Database”
- Chapter 3, “Creating and Configuring an MS SQL Server Database”
- Chapter 4, “Creating and Configuring an IBM DB2 Database”
- Chapter 5, “Creating and Configuring SQL Anywhere”

Chapter 1

Creating and Configuring an Oracle 9 Database

Use this chapter to set up an Oracle 9.2.0.*x* database for your Content Server (Spark) installation. For background information regarding database configuration and users' permissions, see "Creating and Configuring a Database," on page 7.

This chapter contains the following sections:

Step I. Create an Oracle 9.2.0.*x* Database

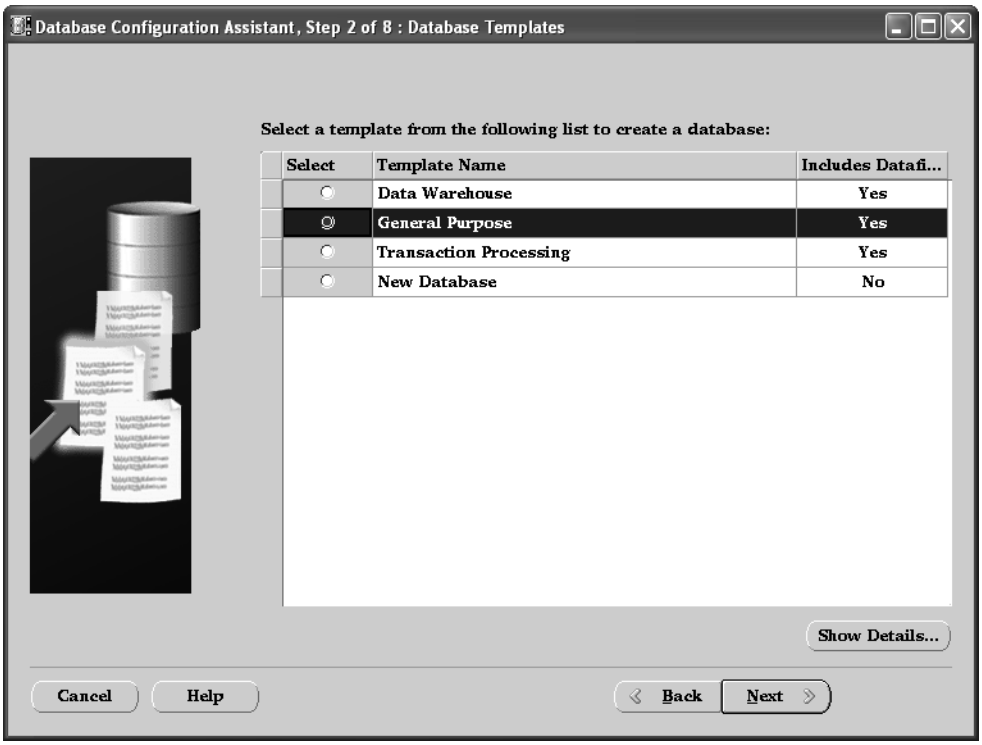
1. Go to the `$ORACLE9_HOME/bin` directory:

   ```
   cd $ORACLE9_HOME/bin
   ```

2. Run the Database Configuration Assistant:

   ```
   dbca
   ```

3. In the welcome screen, click **Next**.
4. Fill in the following screens as shown below:
   a. On the "Step 1 of 8: Operations" screen, leave **Create a database** selected and click **Next**.
   b. On the "Step 2 of 8: Database Templates" screen, select **General Purpose** and click **Next**.



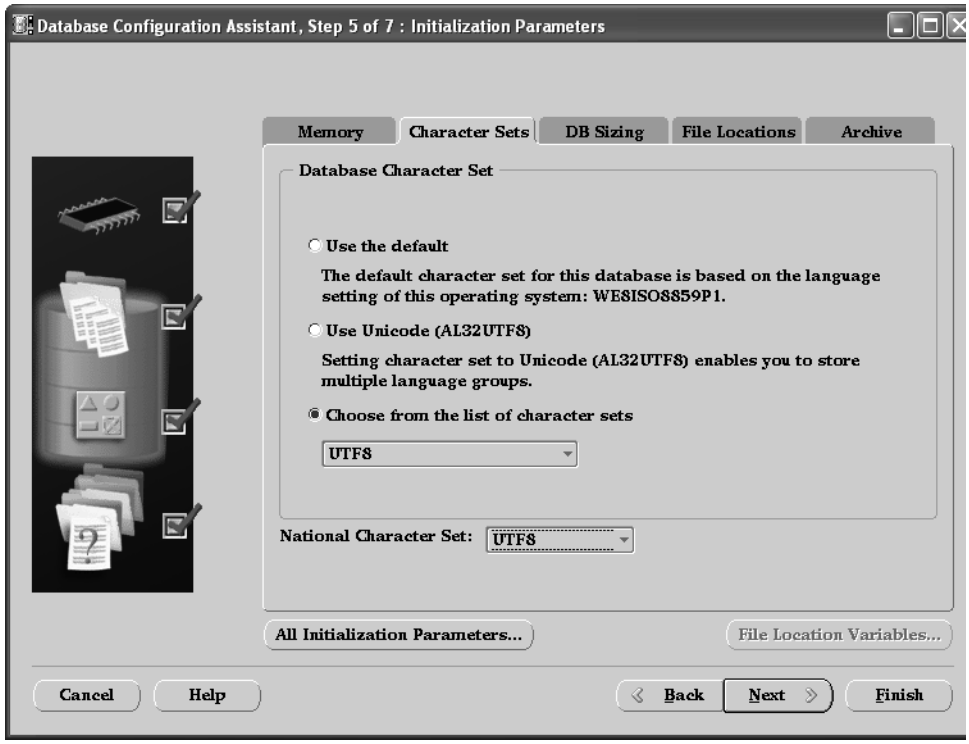
c. On the "Step 3 of 7: Database Identification" screen, enter the database name in the Global Database Name field. The SID will be automatically set to the first eight characters of the Database Name. Each SID must be unique. Click **Next**.

d. On the "Step 4 of 7: Database Connection Options" screen, leave **Dedicated Server mode** selected and click **Next**.

e. On the "Step 5 of 7: Initialization Parameters" screen:

1) Select the tab **Character Sets**.

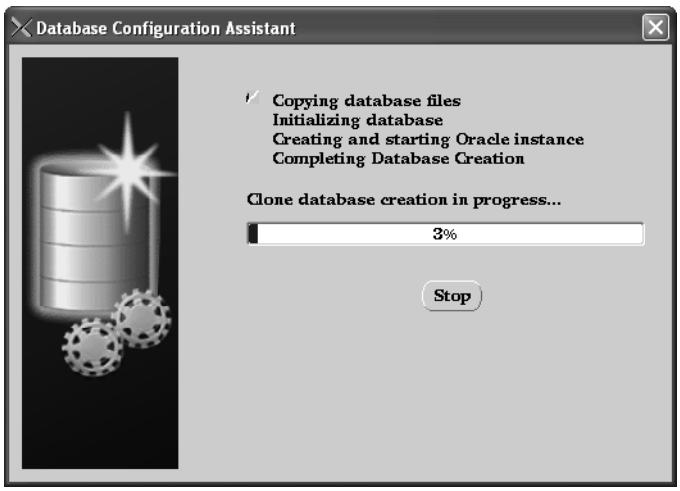
2) On the character sets screen, select **Choose from the list of character sets** and **UTF8** from the drop-down menu. For the National Character Set select **UTF8** and click **Next**.



f. On the "Step 6 of 7: Database Storage" screen, click **Next**.

g. On the "Step 7 of 7: Creation Options" screen, click **Finish**.

5. When the summary screen appears, click **OK**.
6. When the "Installation Progress Screen" appears, wait for the installation to be completed.

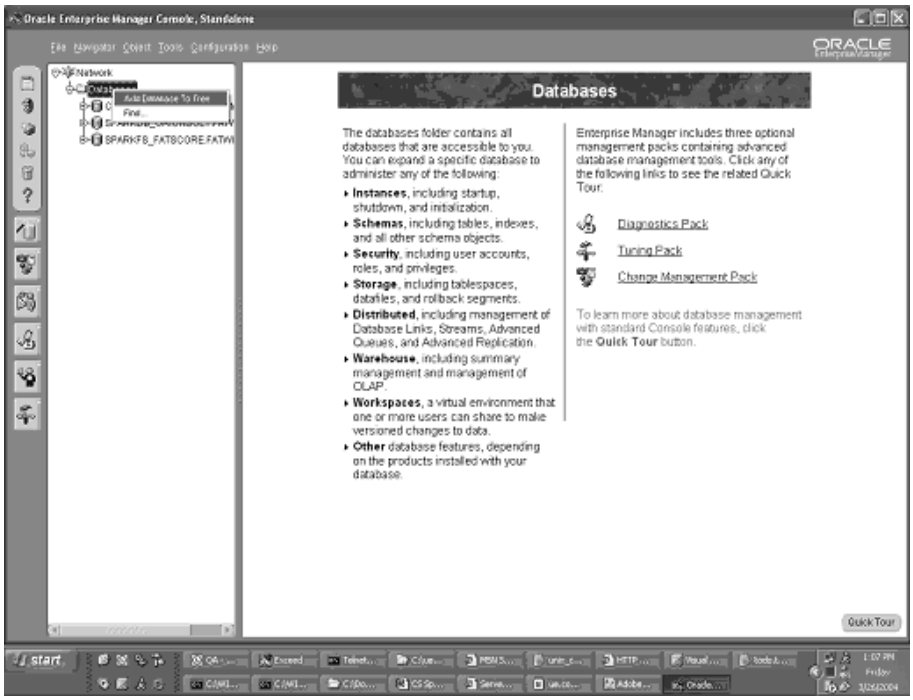


7. After the installation has been completed, the database information screen will appear, listing information about how to connect to this database using the enterprise management console. Click **OK**.

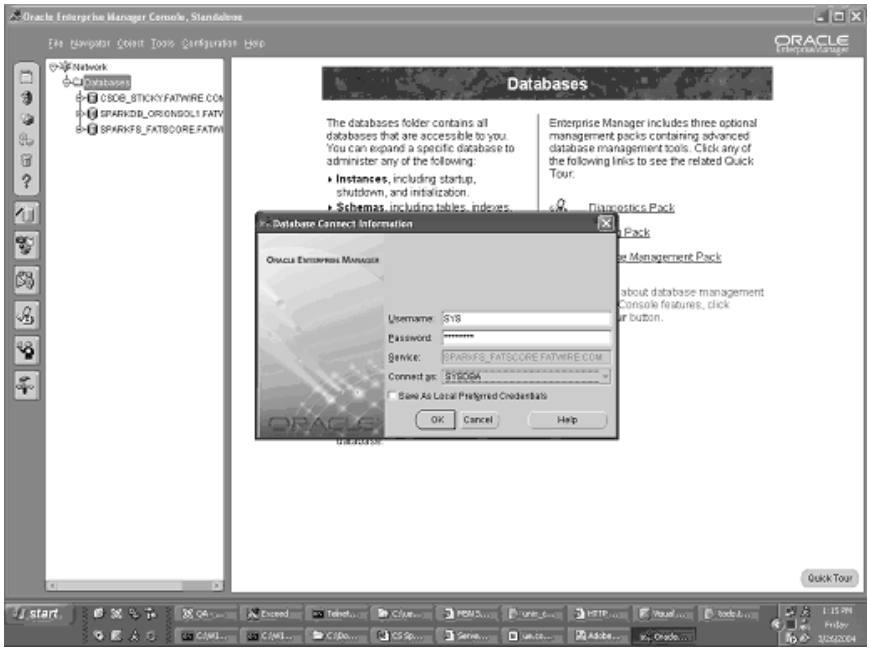
Step II. Configure the Database for Content Server (Spark)

1. Start Oracle Enterprise Manage Console. Click **OK**.

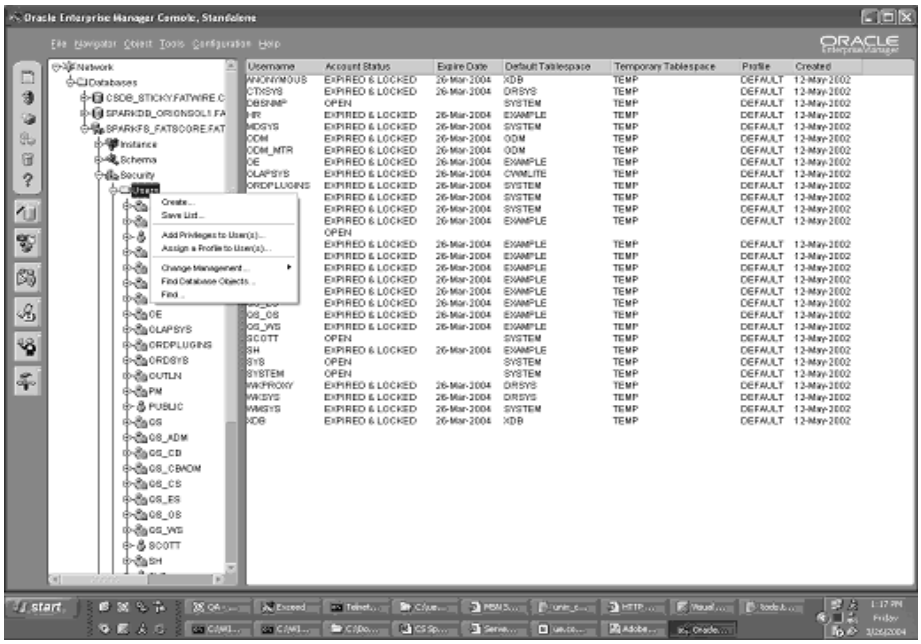
2. Expand **Network > Databases**. Select the database you have created.



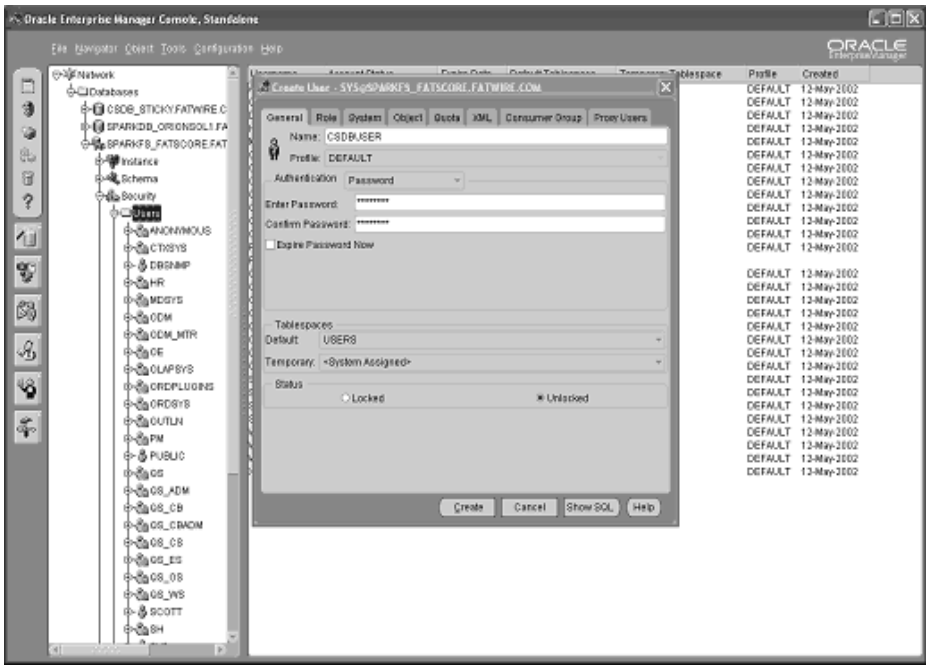
3. In the database login screen, log in as SYS and enter the password you specified earlier while creating the database. Click **OK**.



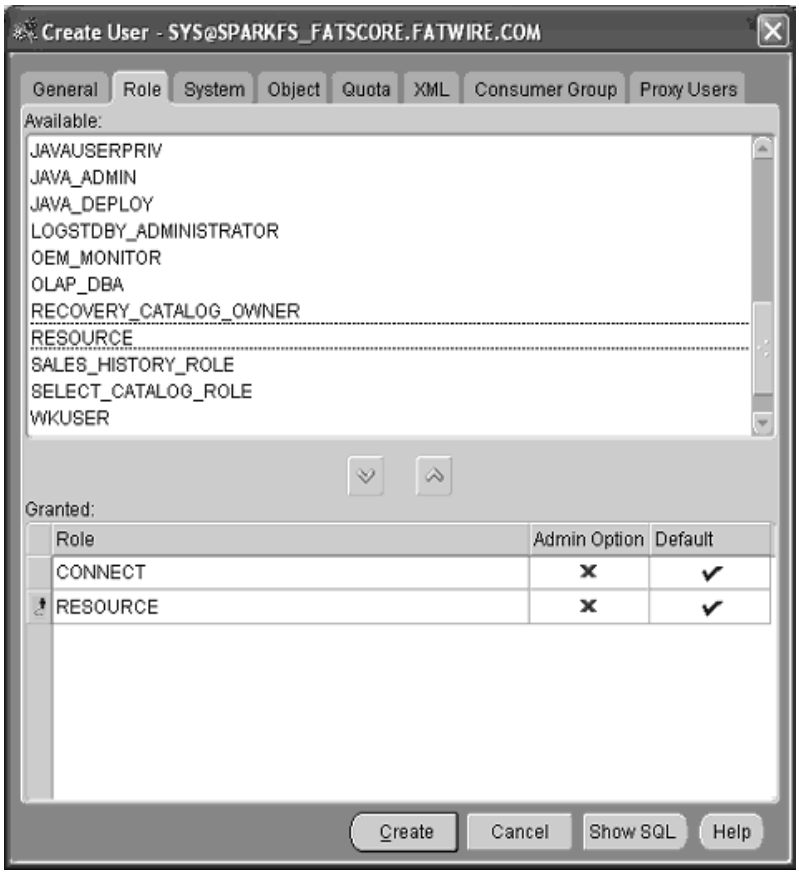
4. Expand **Security > Users**. Right-click and select **Create**.



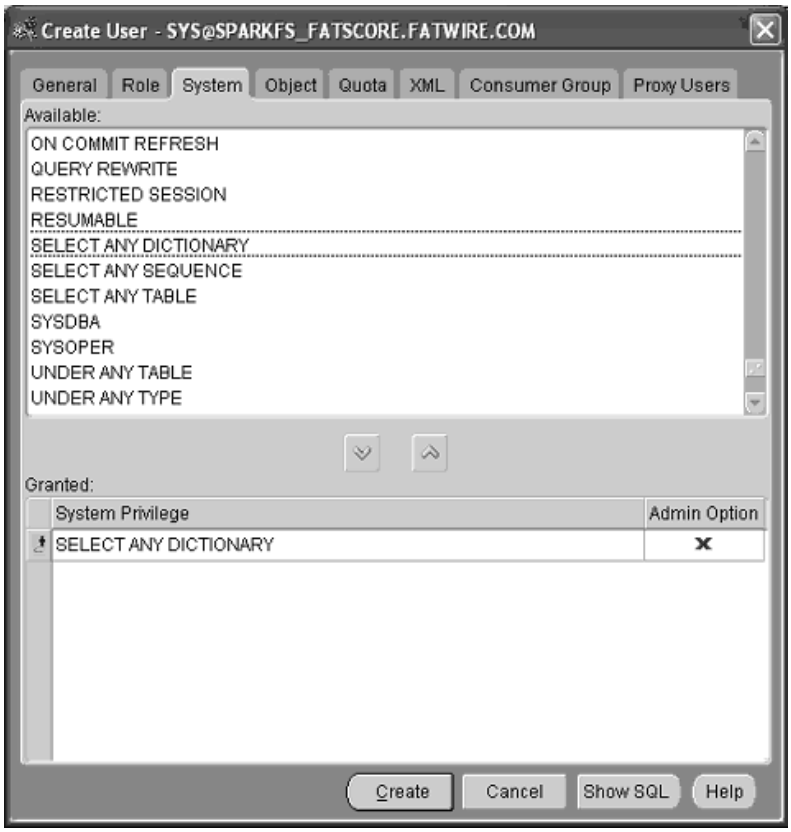
5. On the **General** tab, enter a username and password.



6. On the **Role** tab, select **Connect** and **Resource**.



7. On the **System** tab, choose **Select Any Dictionary**. Click **Create**.



8. Database configuration is complete. You are now ready to create and configure the data source. For instructions, refer to your Content Server (Spark) installation guide.

Chapter 2

Creating and Configuring an Oracle 10*g* Database

Use this chapter to set up an Oracle 10*g* database for your Content Server (Spark) installation. For background information regarding database configuration and users' permissions, see Part 1, "Creating and Configuring a Database."

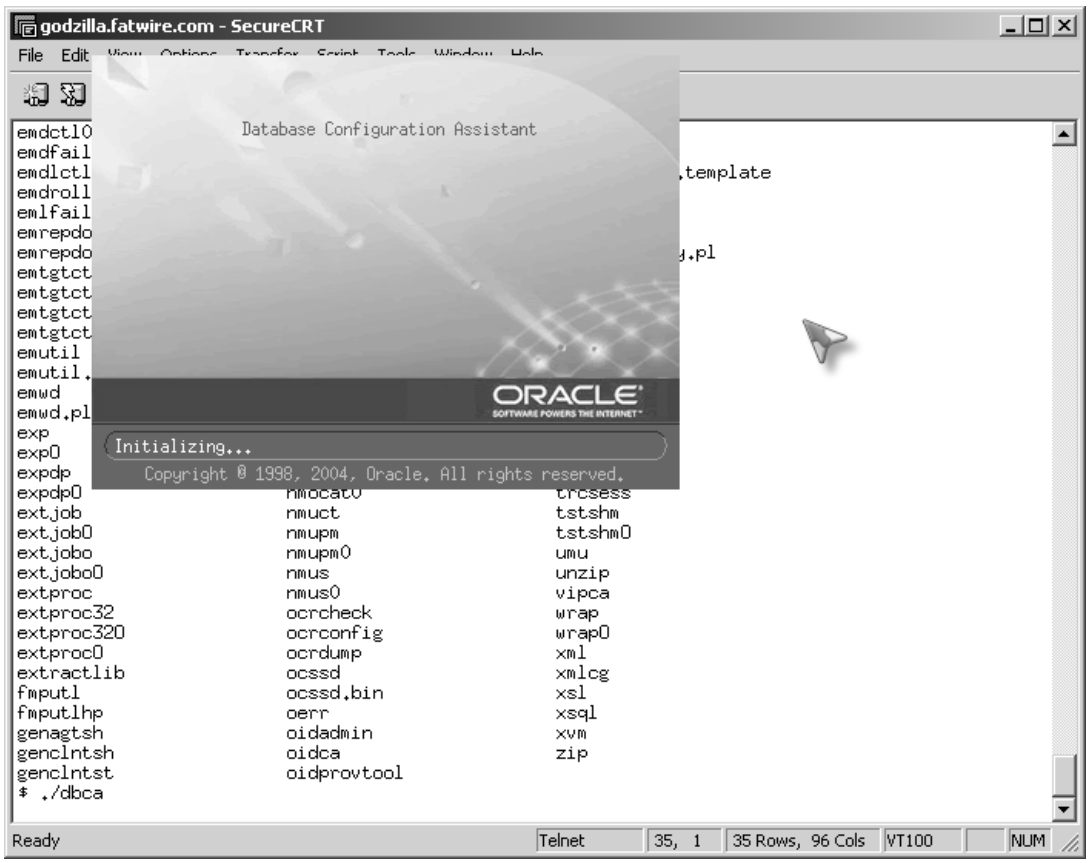
This chapter contains the following sections:

- Step I. Create an Oracle 10g Database
- Step II. Create a New User for Content Server (Spark)
- Step III. (For Type 2 Drivers) Install and Configure an Oracle Client

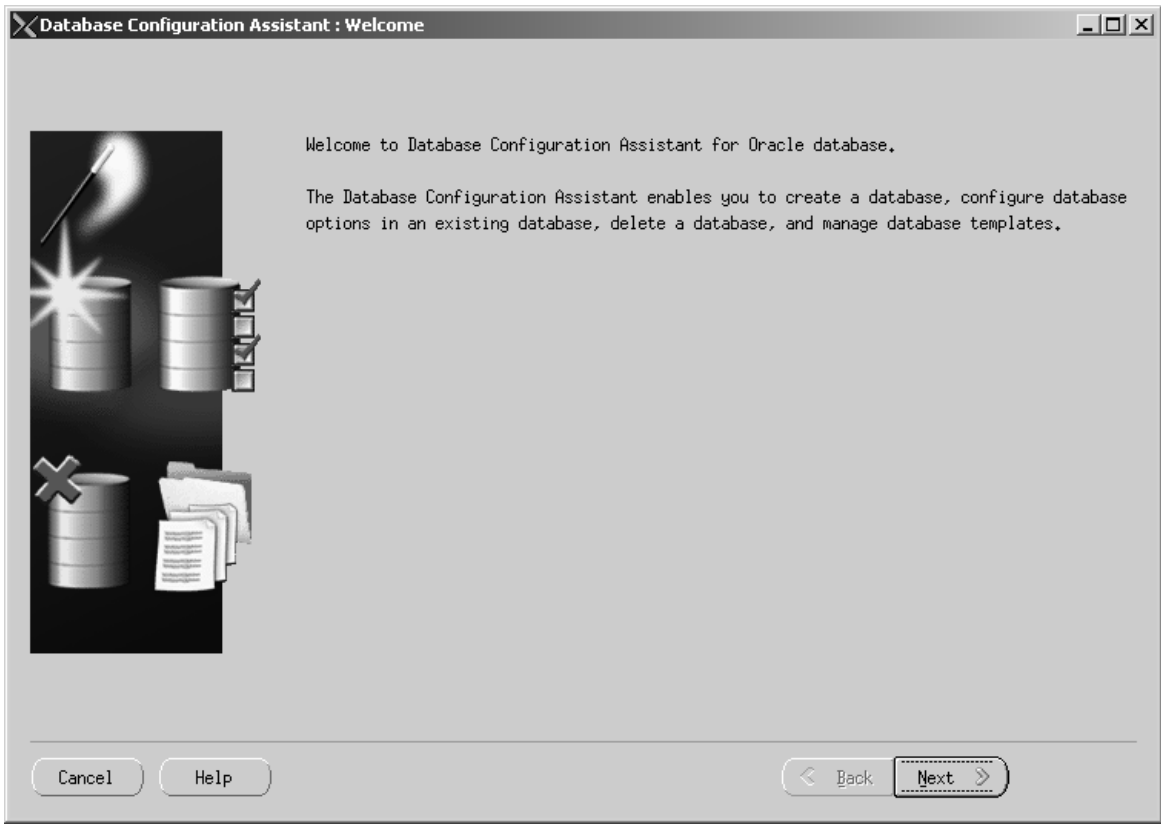
Step I. Create an Oracle 10*g* Database

1. Execute the "Oracle Database Configuration Assistant" by doing one of the following:
   - In Unix, execute the command: **dbca**
   - In Windows, go to the "Oracle Programs" group and select **Database Configuration Assistant**.

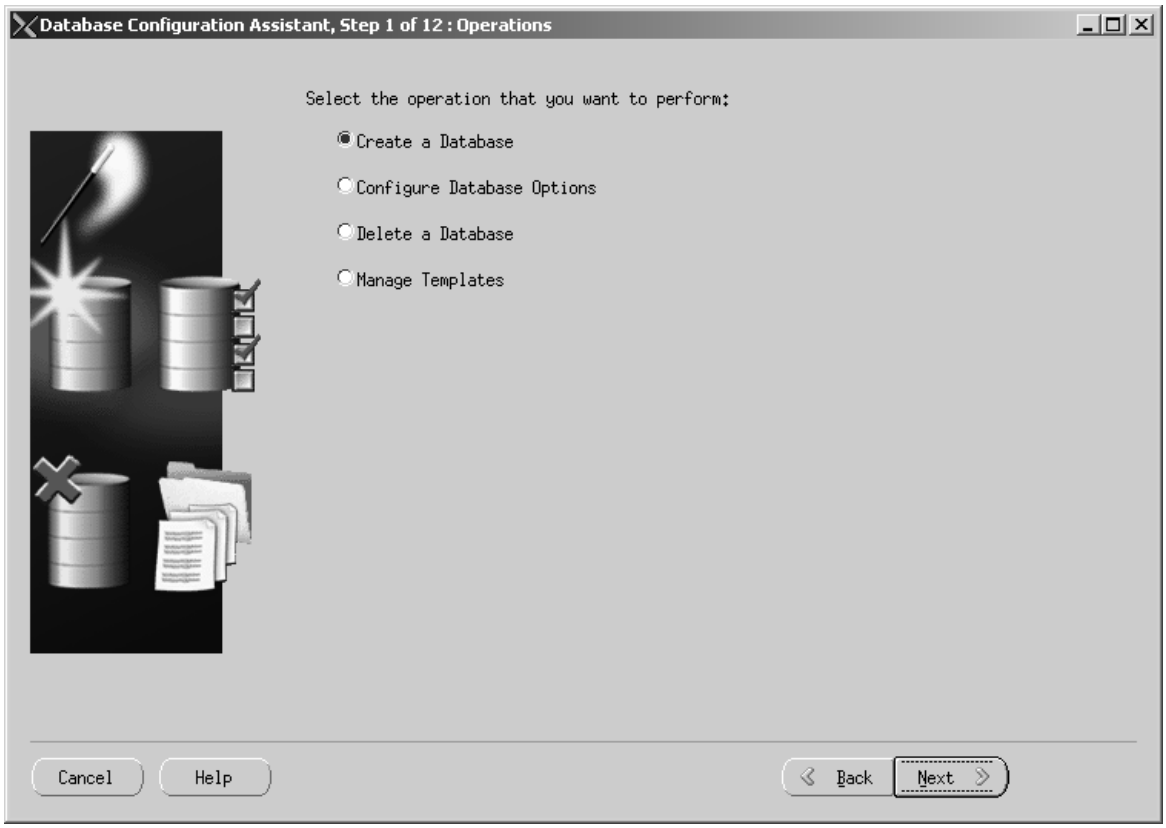
> **Note**
>
> This step displays a load screen that can take some time to complete. Be patient.



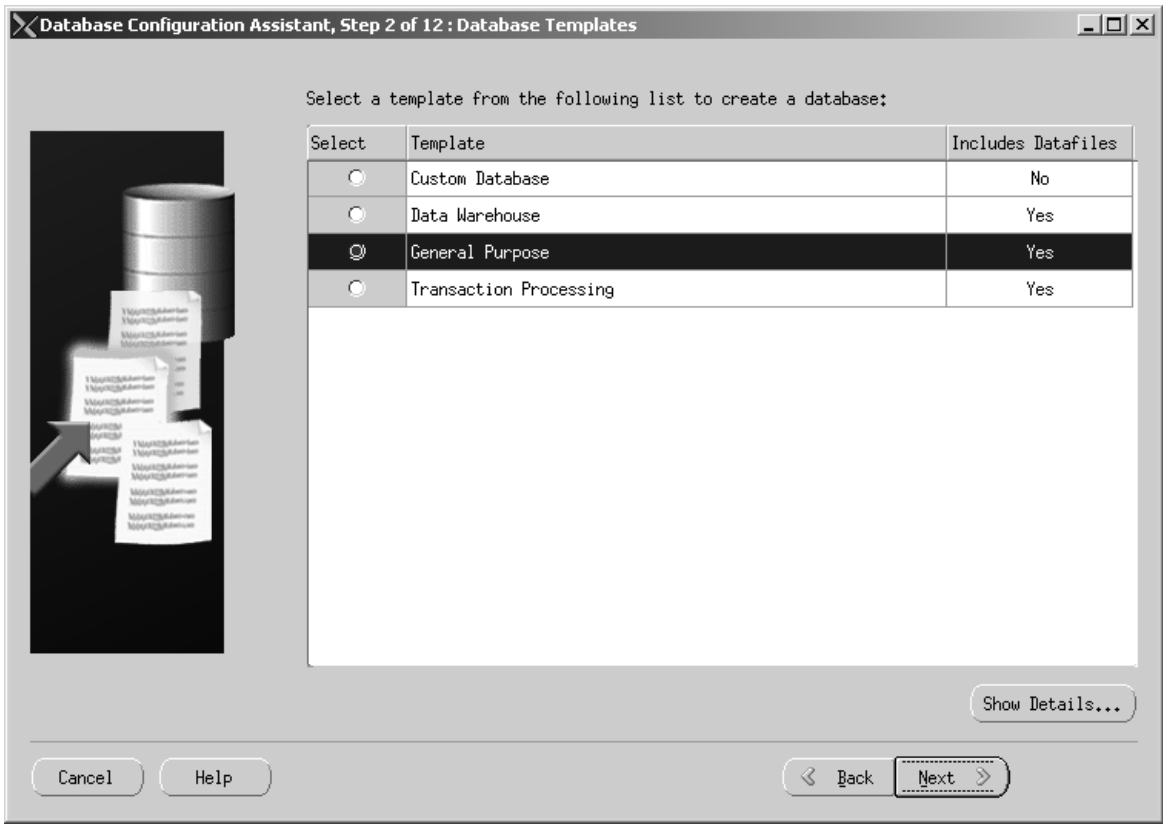
2. On the first screen that is displayed, click **Next**. In the following screen, click **Next**.



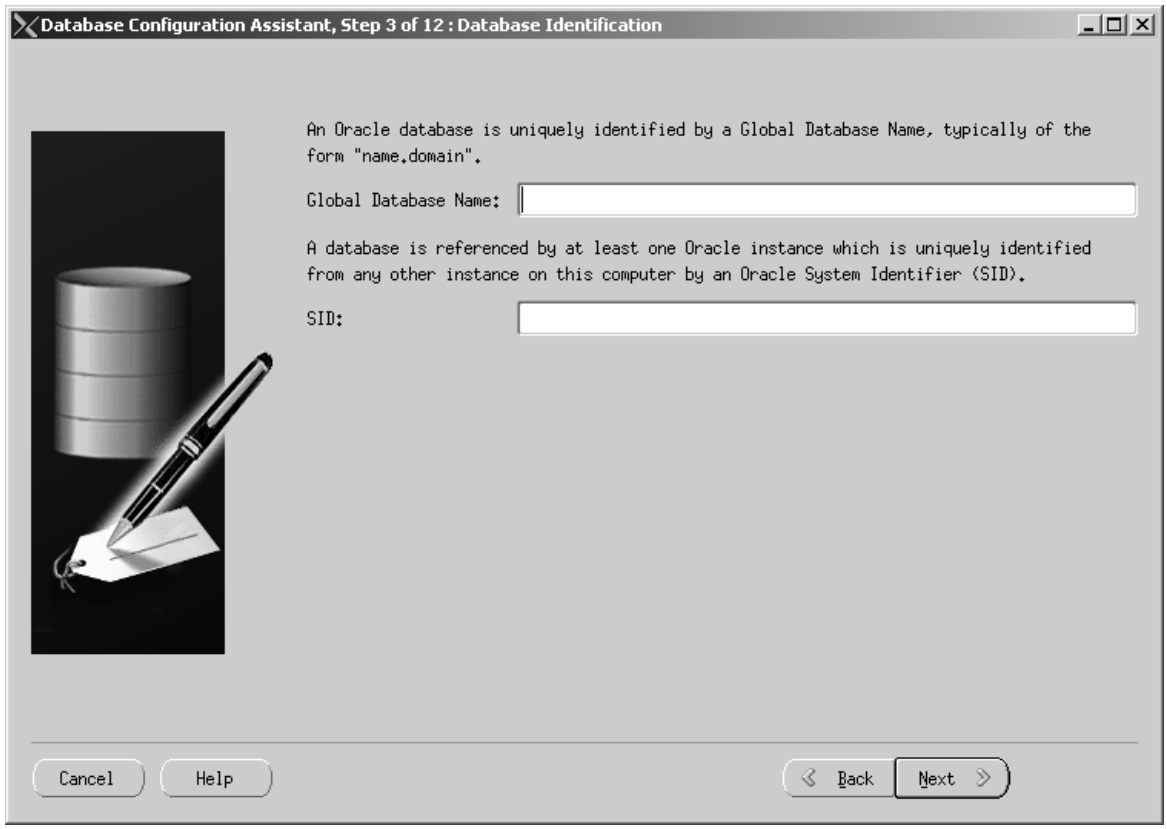
3. Select the radio button **Create a Database** and click **Next**.



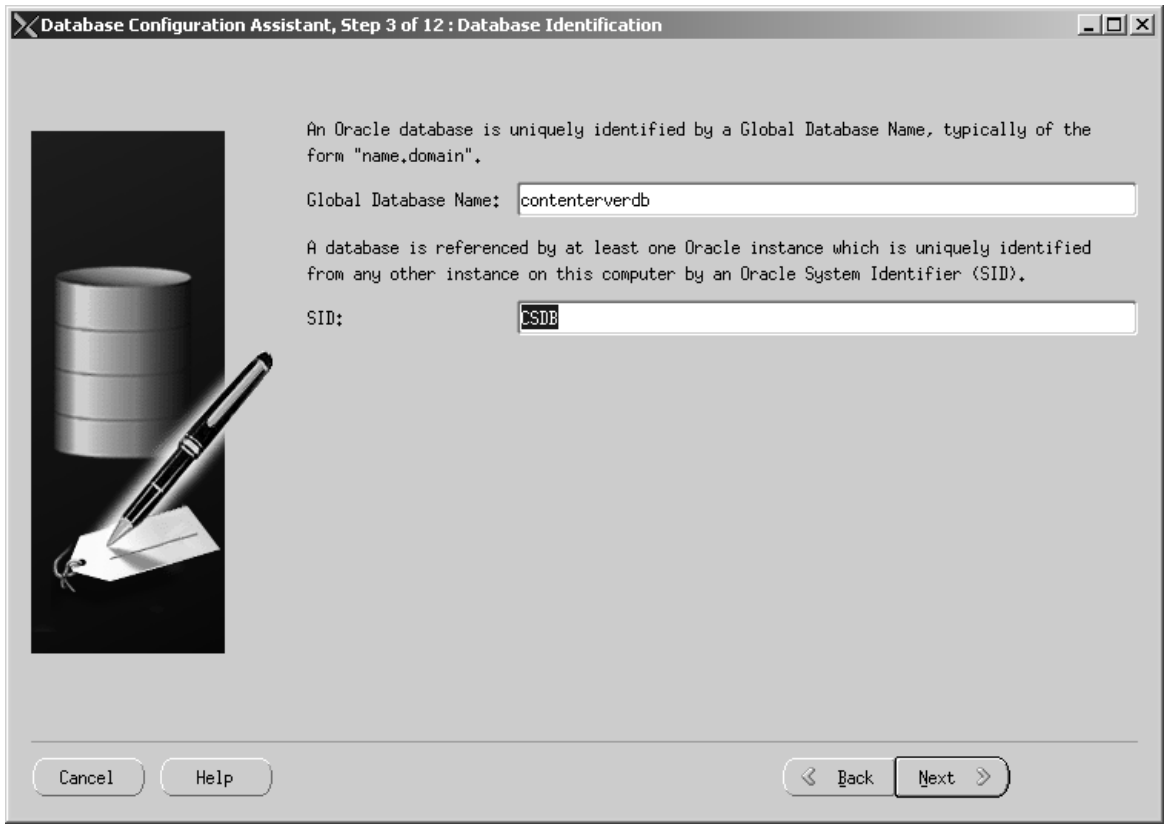
4. Select the radio button **General Purpose** and click **Next**.



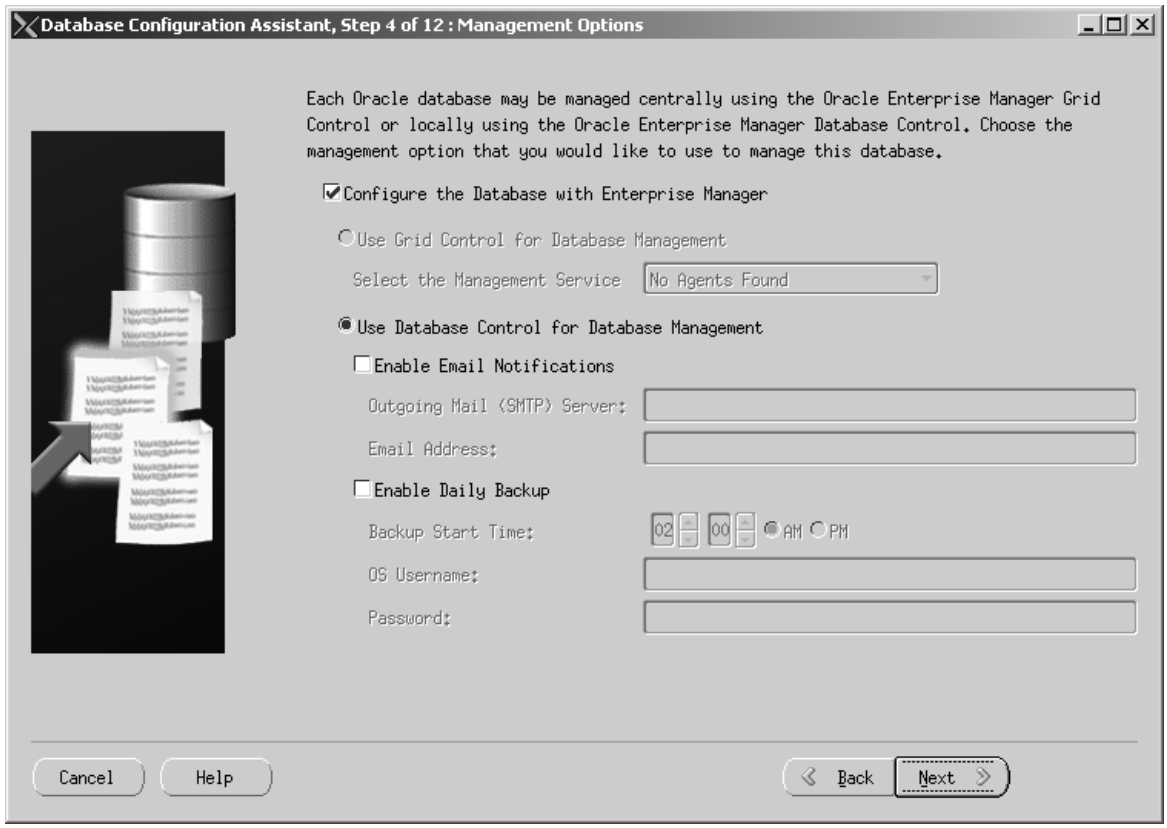
5. Enter a unique global database name and SID (in this example the global database name is `contentserverdb`. The SID is `CSDB`). Click **Next**.



6. Do not change any options. Click **Next**.

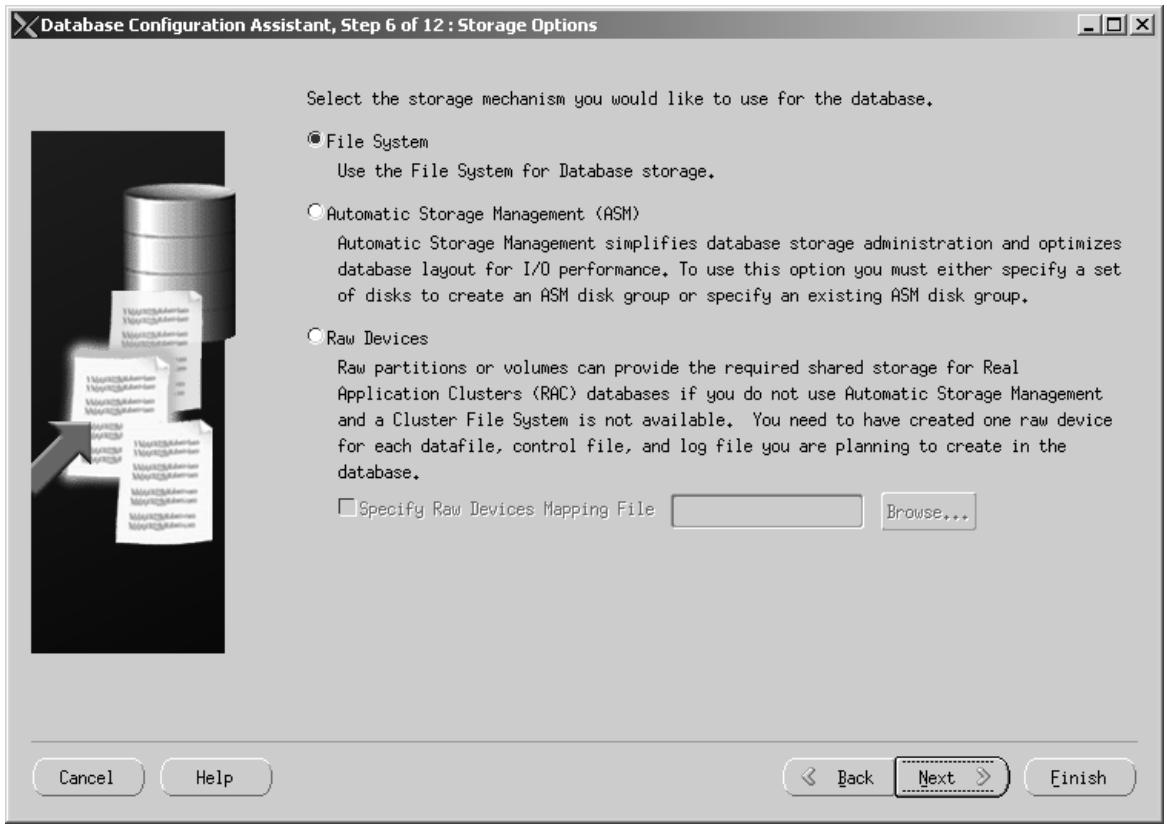


7. Do not change any options. Click **Next**.

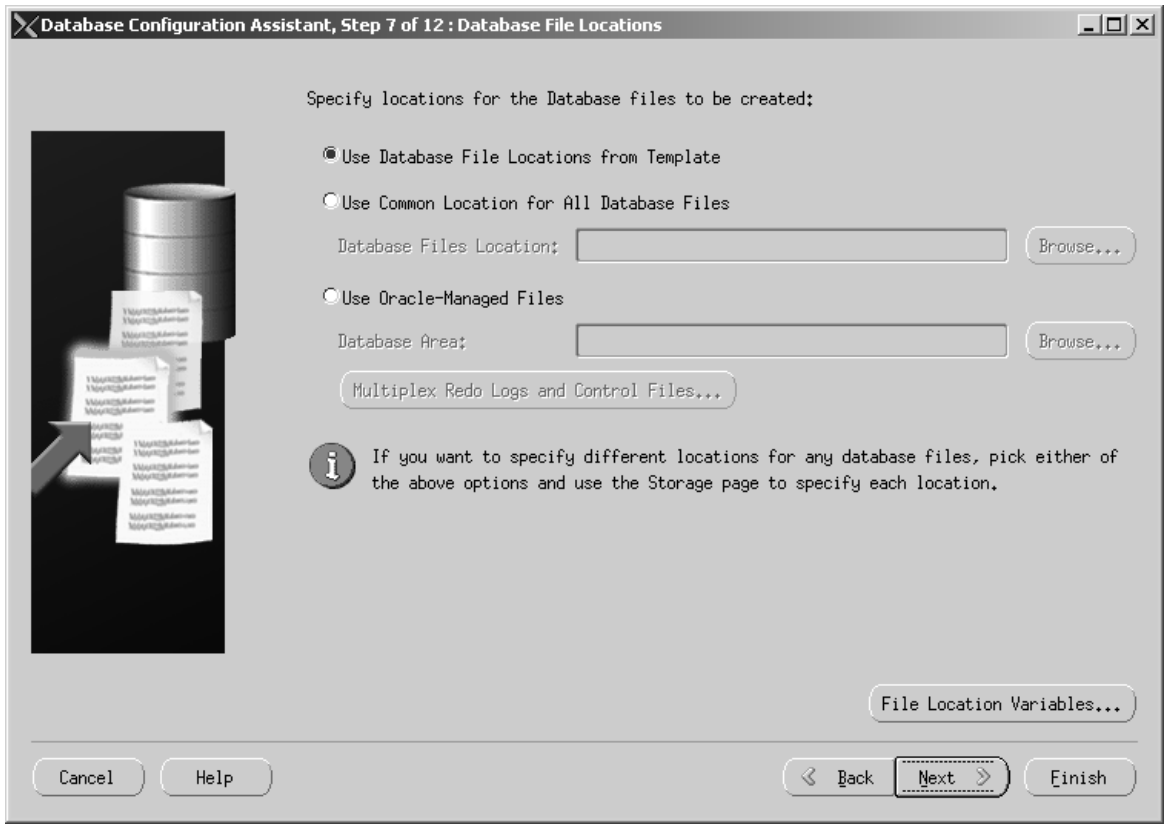


8. Enter a password, re-enter the same password in the "Confirm Password" field and click **Next**.
9. For enhanced security select the radio button **Use Different Passwords** and enter a unique password for each of the given users.

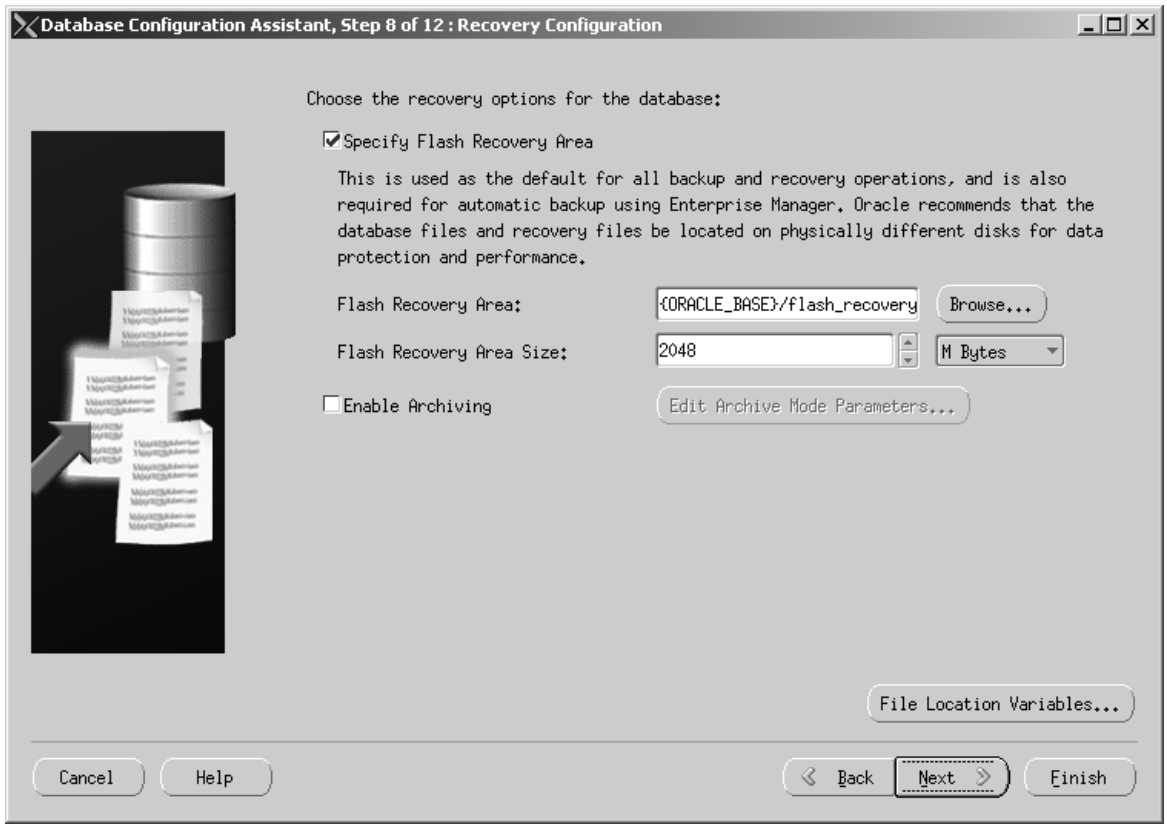
10. Do not change any options. Click **Next**.



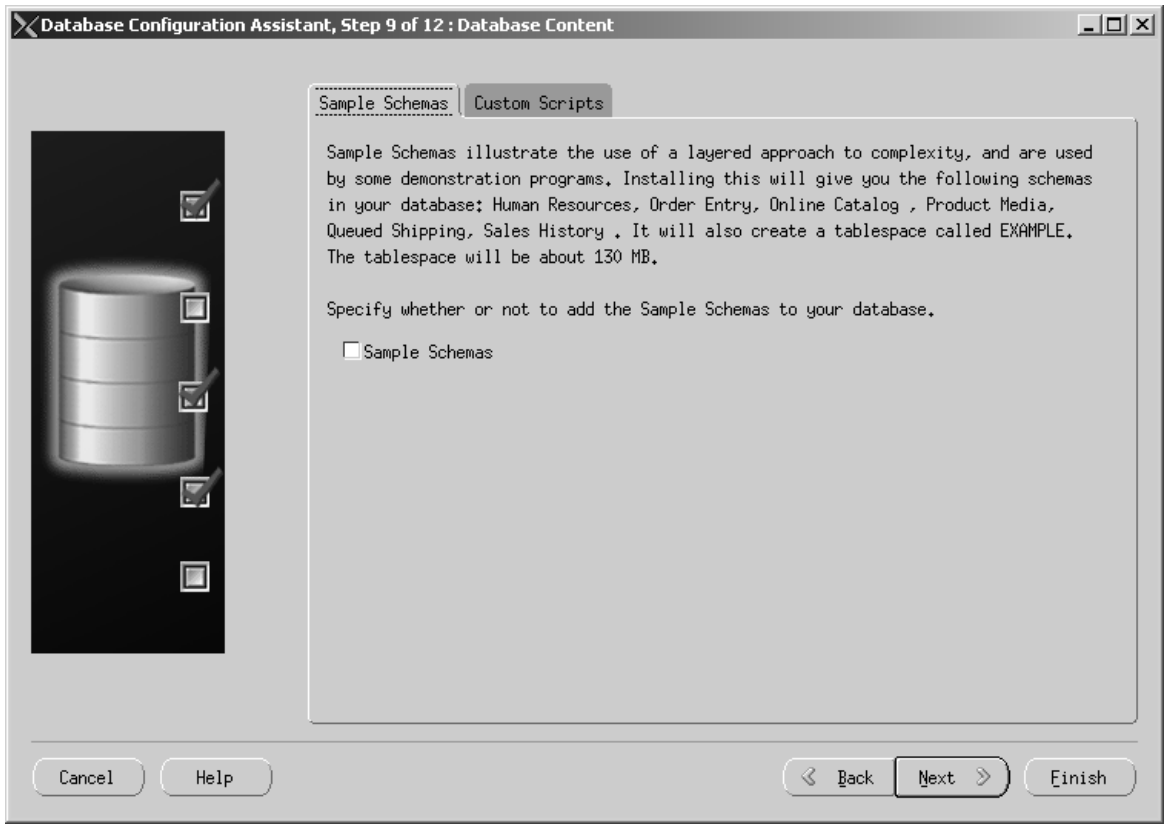
11. Do not change any options. Click **Next**.



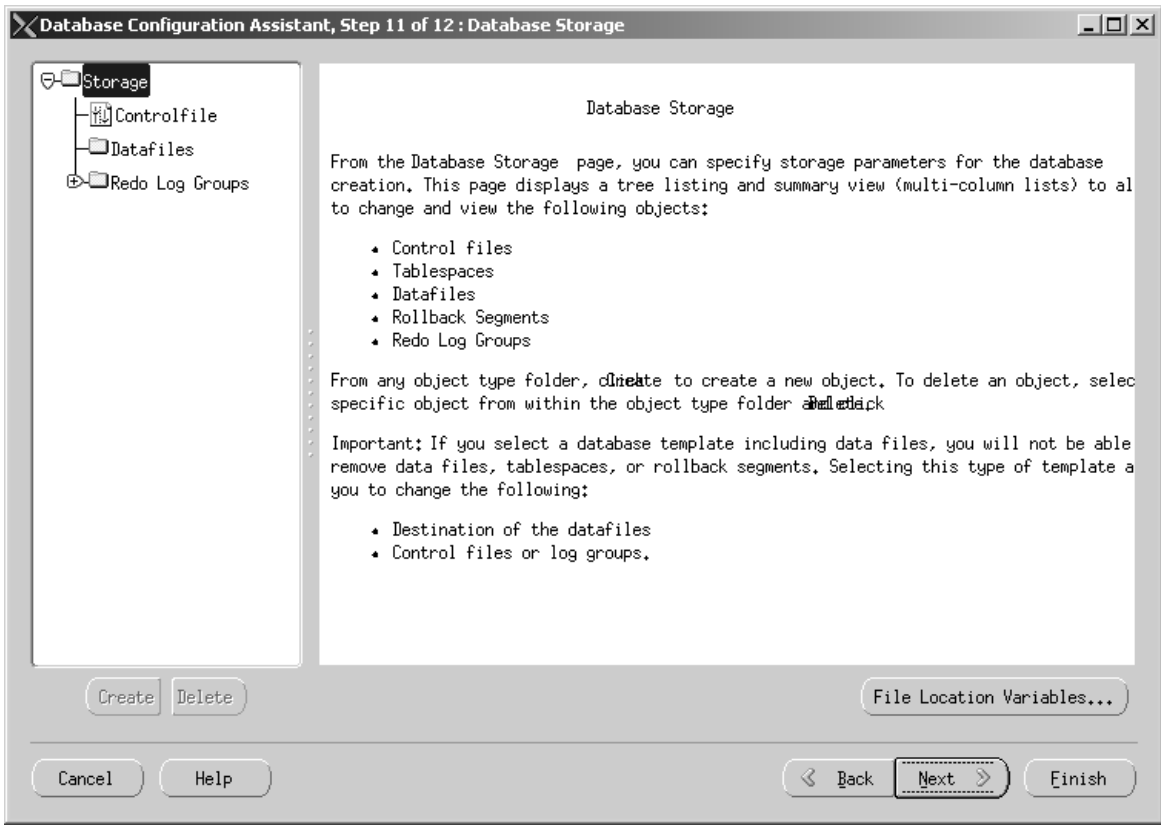
12. Do not change any options. Click **Next**.



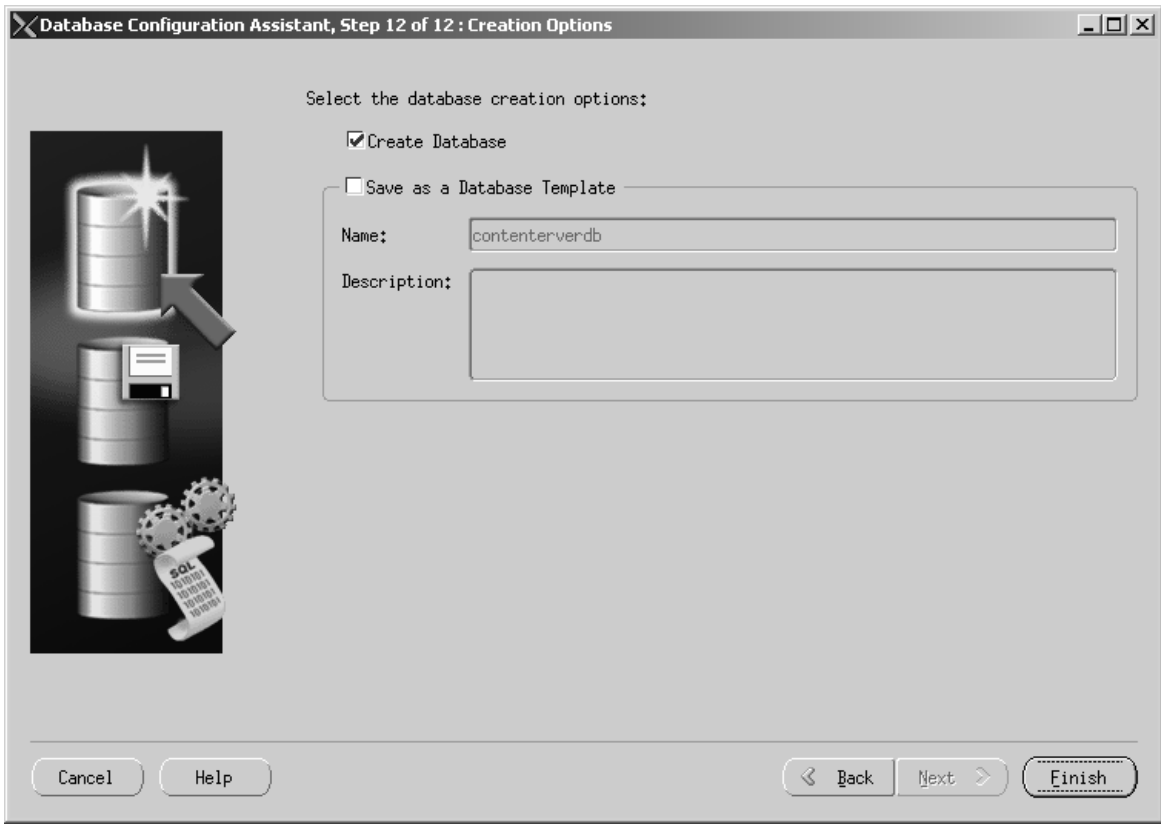
13. Do not change any options. Click **Next**.



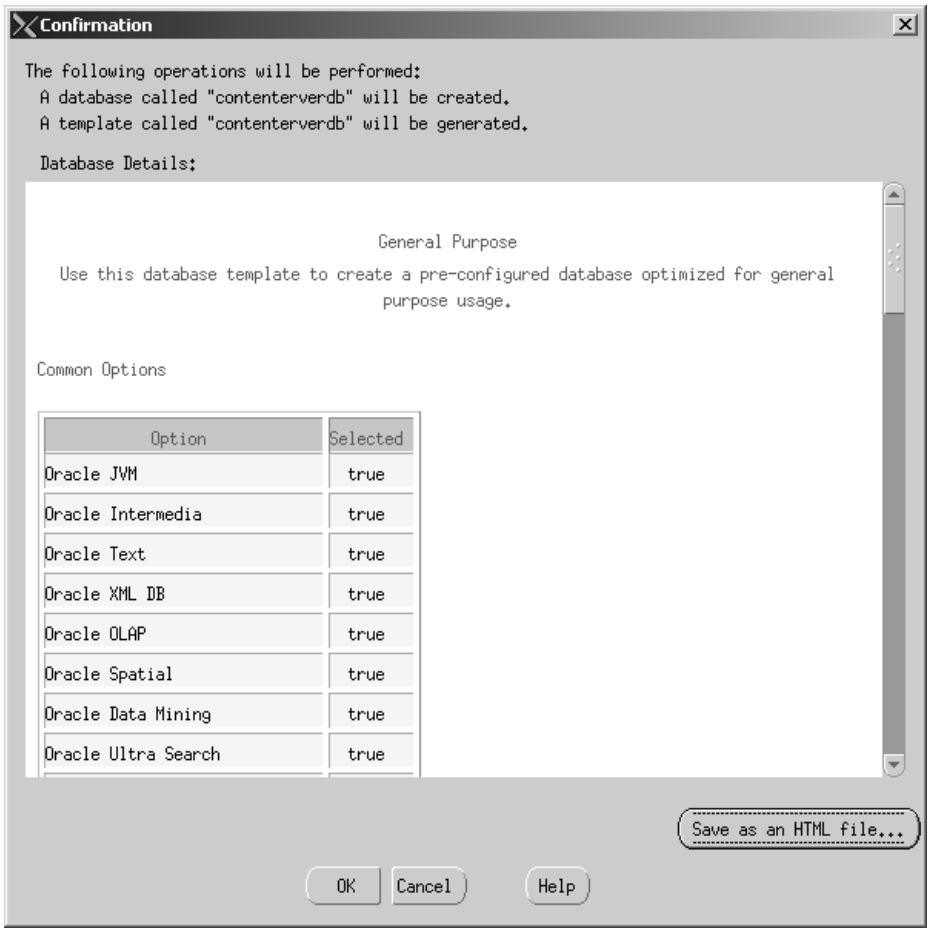
14. Click the **Character Sets** tab and do the following:
    a. Select **Choose from the list of character sets** and select **UTF-8** from the drop-down menu.
    b. Click the **National Character Set** drop down-menu and select **UTF8**.
15. Leave all other options on the different tabs as is and click **Next**.
16. For database storage, no options need to be changed. However, if you wish to change the location of the database from the default of `oradata` located under the Oracle installation, you can do so on this page. Click **Next**.



17. Do not change any options. Click **Finish**.



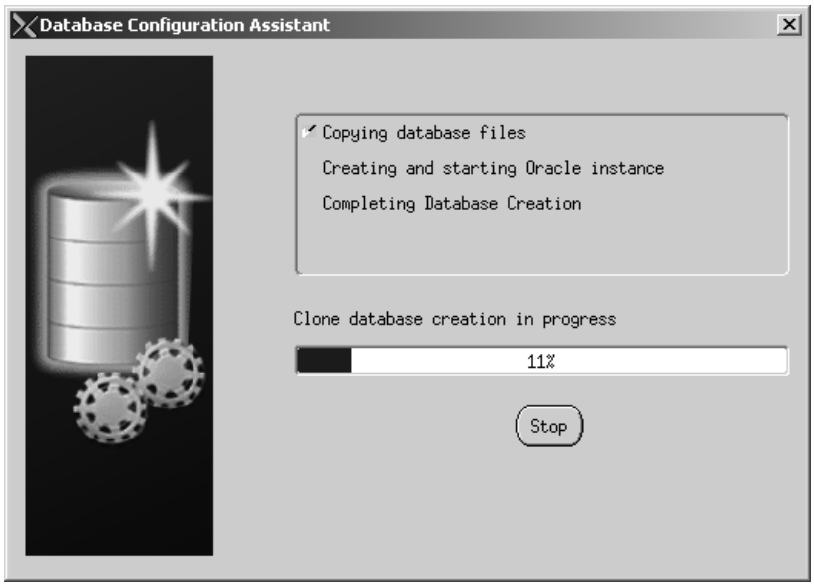
18. In the "Configuration" window, review the choices that you made on the previous screens. If you need to modify your choices, click **Cancel** and make the modifications. Otherwise, click **OK** to continue.



Note

If you are planning to use internationalization, for Content Server (Spark) the key value is: `National Character Set: UTF8`

19. The following window shows the progress of the database creation. This step can take time to complete.



20. When database creation is completed, click **Exit**.

Step II. Create a New User for Content Server (Spark)

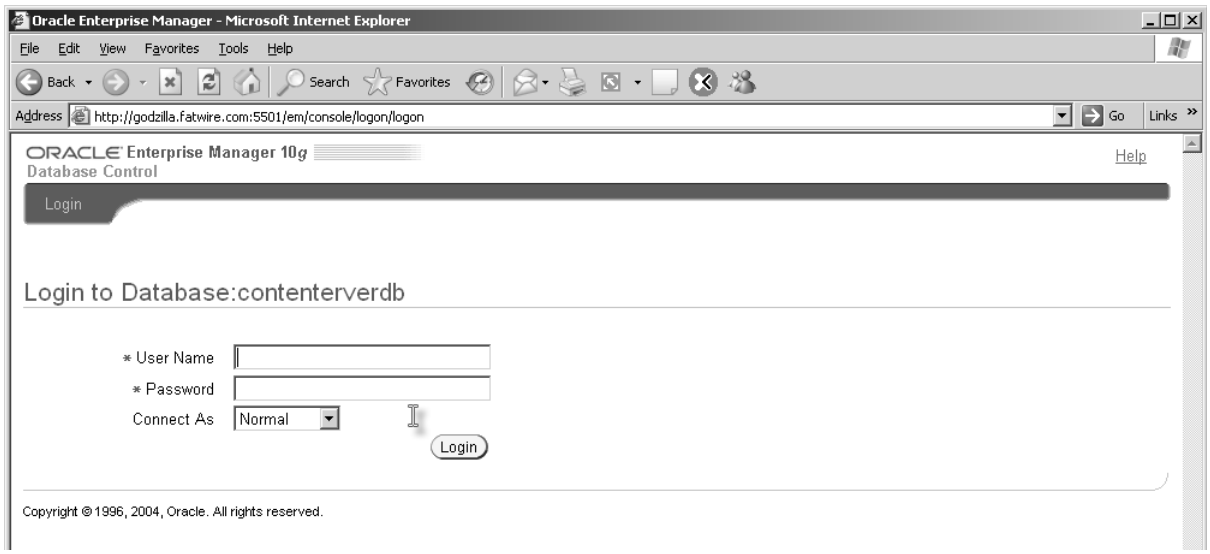
1. Locate the file `emoms.properties` (in `<oracle home>/<server name>_<SID>/sysman/config/`).
   a. Find the line: `oracle.sysman.emSDK.svlt.ConsoleServerPort`
   b. The port after the line in step a is important. Make a note of it.
2. Run the command: **`emctl status dbconsole`**

   The command should return an output similar to the following:

```
Oracle Enterprise Manager 10g Database Control Release
     10.1.0.2.0
Copyright (c) 1996, 2004 Oracle Corporation. All rights
     reserved.
http://godzilla:5500/em/console/aboutApplication
     Oracle Enterprise Manager 10g is running.
------------------------------------------------------------------
Logs are generated in directory /u02/app/oracle10/product/
     10.1.0/db_1/godzilla_orcl10so/sysman/log
```

> **Note**
>
> If the command returns the message that the Oracle Enterprise Manager is not running, start Oracle Enterprise Manager with the command: **`emctl start dbconsole`**

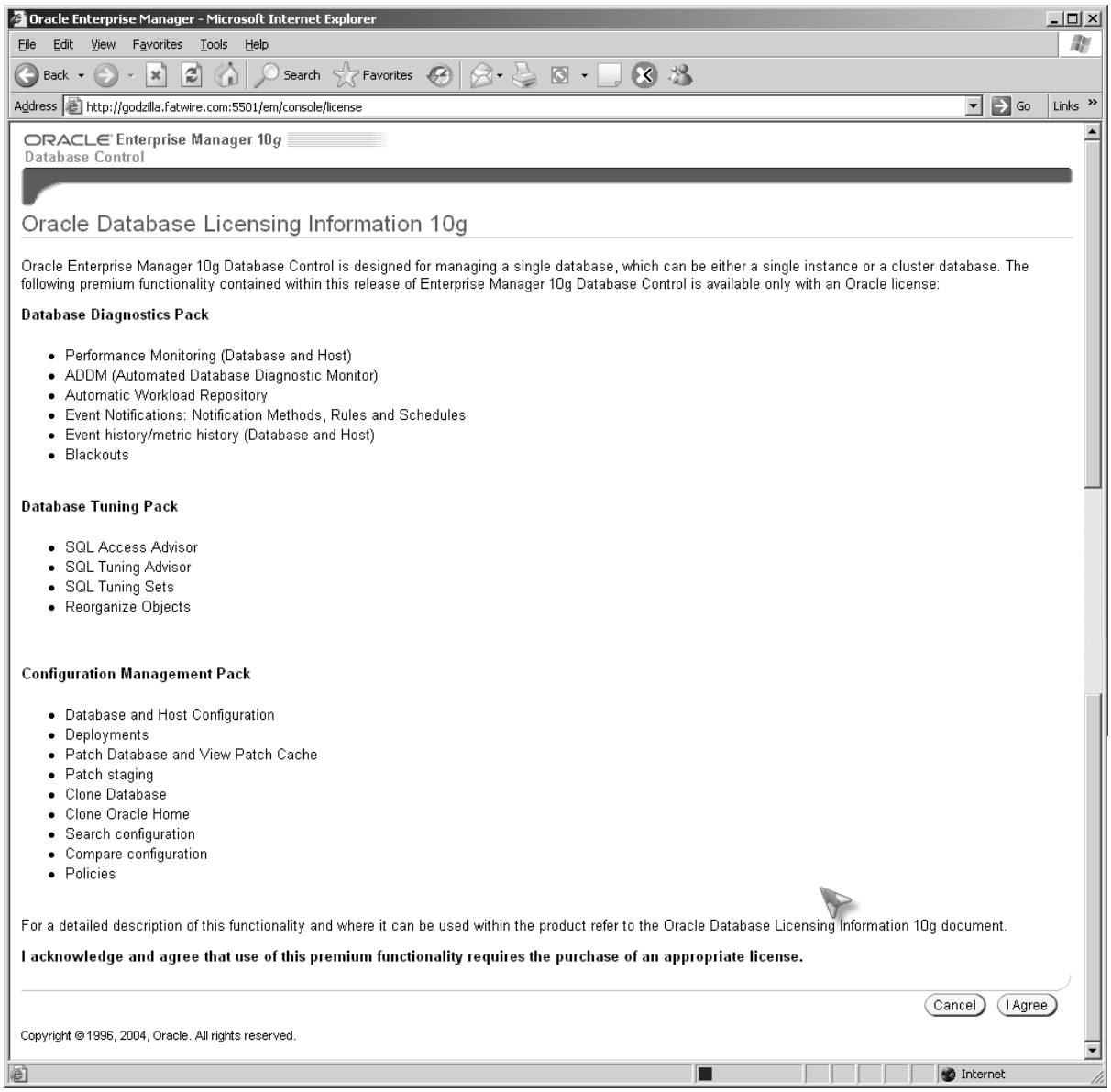
3. Open a browser and do the following:
   a. Browse to the URL `http://<hostname>:<port>/em` (from step 2).



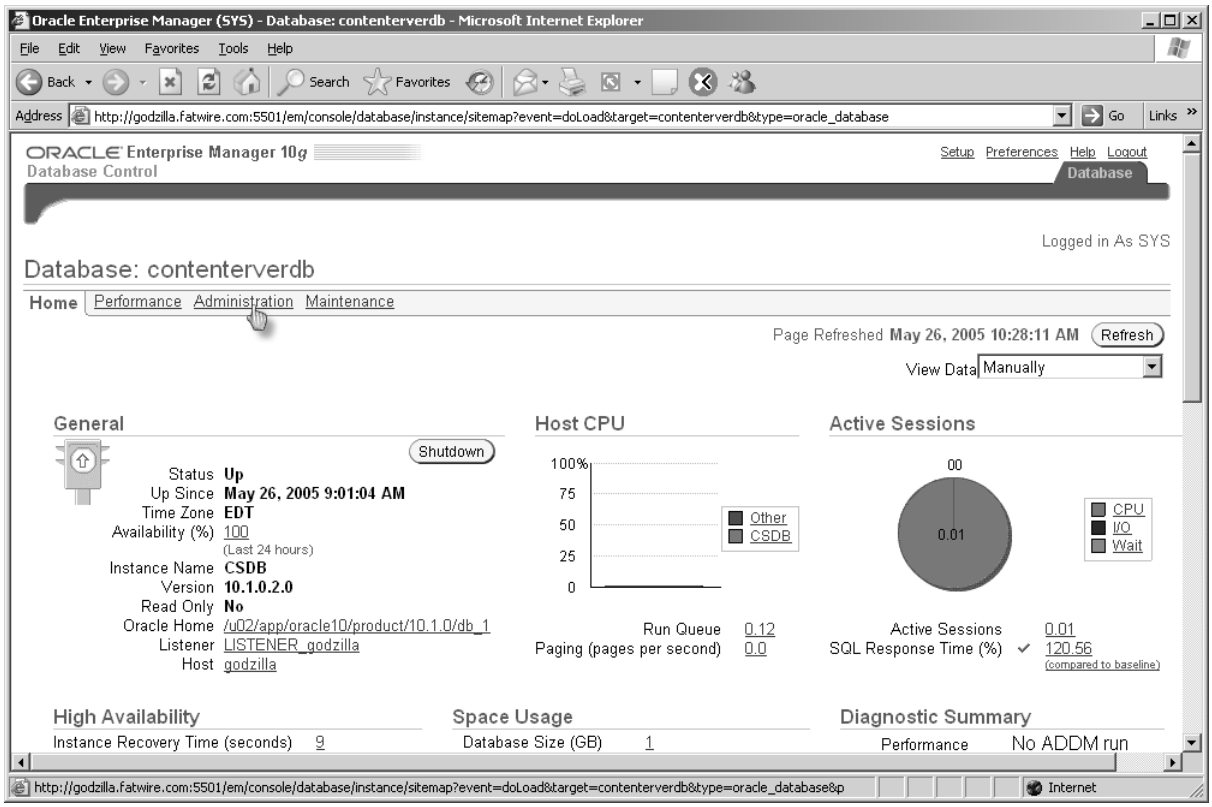
b. Log in to the browser, using the following field values:

Field Name	Field Value
User name	`sys`
Password	`<password entered when creating the db>`
Connect As	`SYSDBA`

c. As this is the first time you are using the Oracle Enterprise Manager, a license page is displayed. Click **I Agree**.



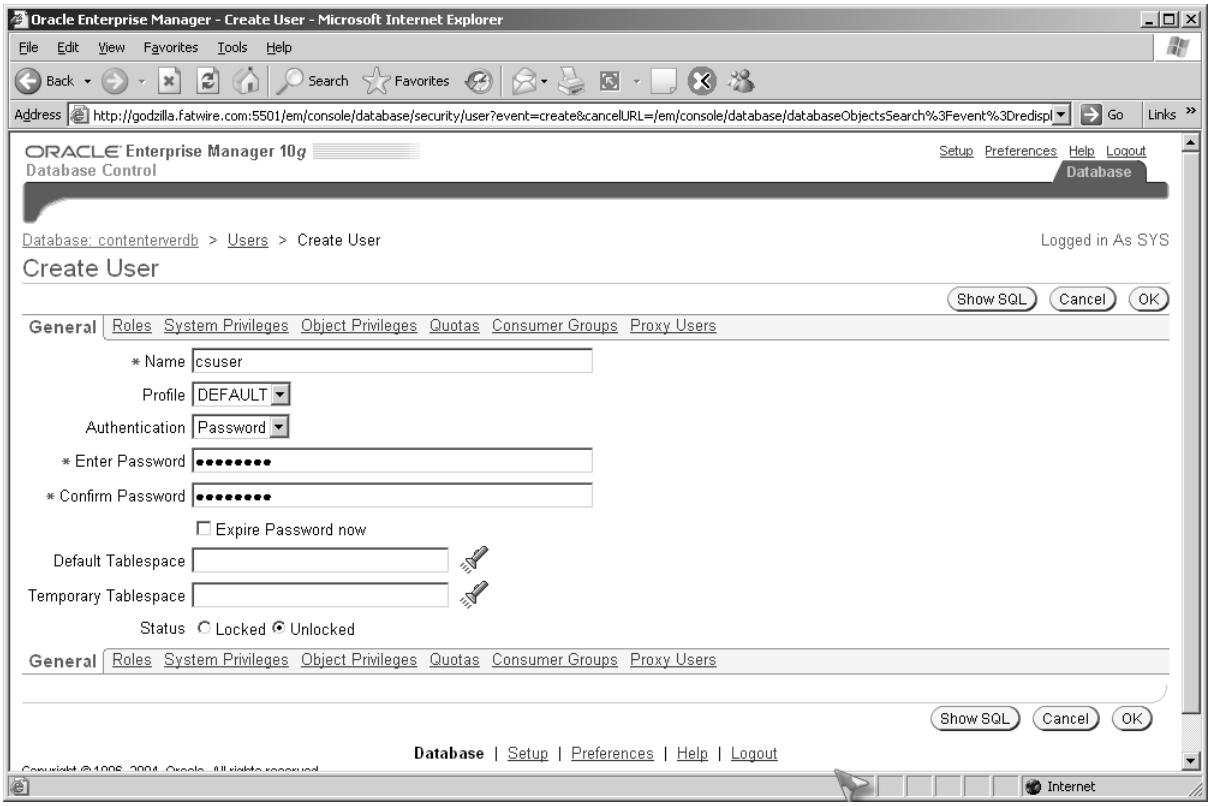
4. Click the **Administration** tab.



a. From the **Security** menu, select **Users**. Click the **Create** button.

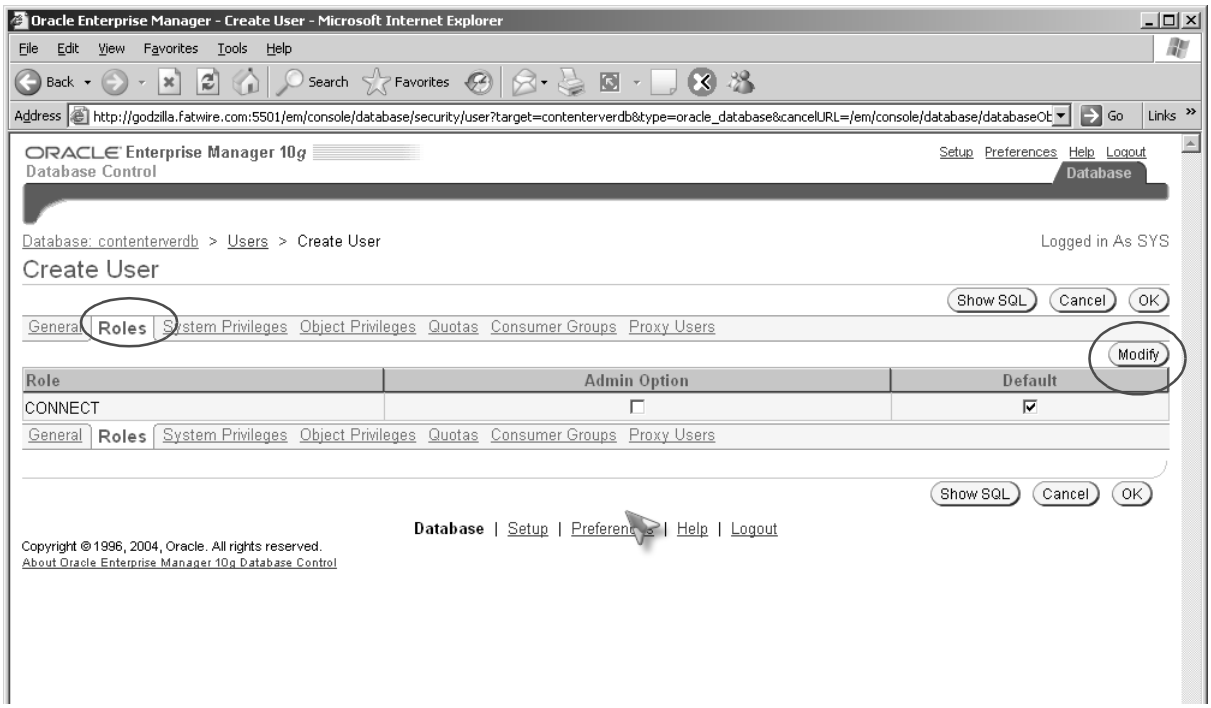
b. In the "Create User" screen, fill in required fields with the values that are listed in the following table:

Field Name	Field Value
Name	`csuser`
Enter Password	`<your choice>`
Confirm Password	`<same password>`

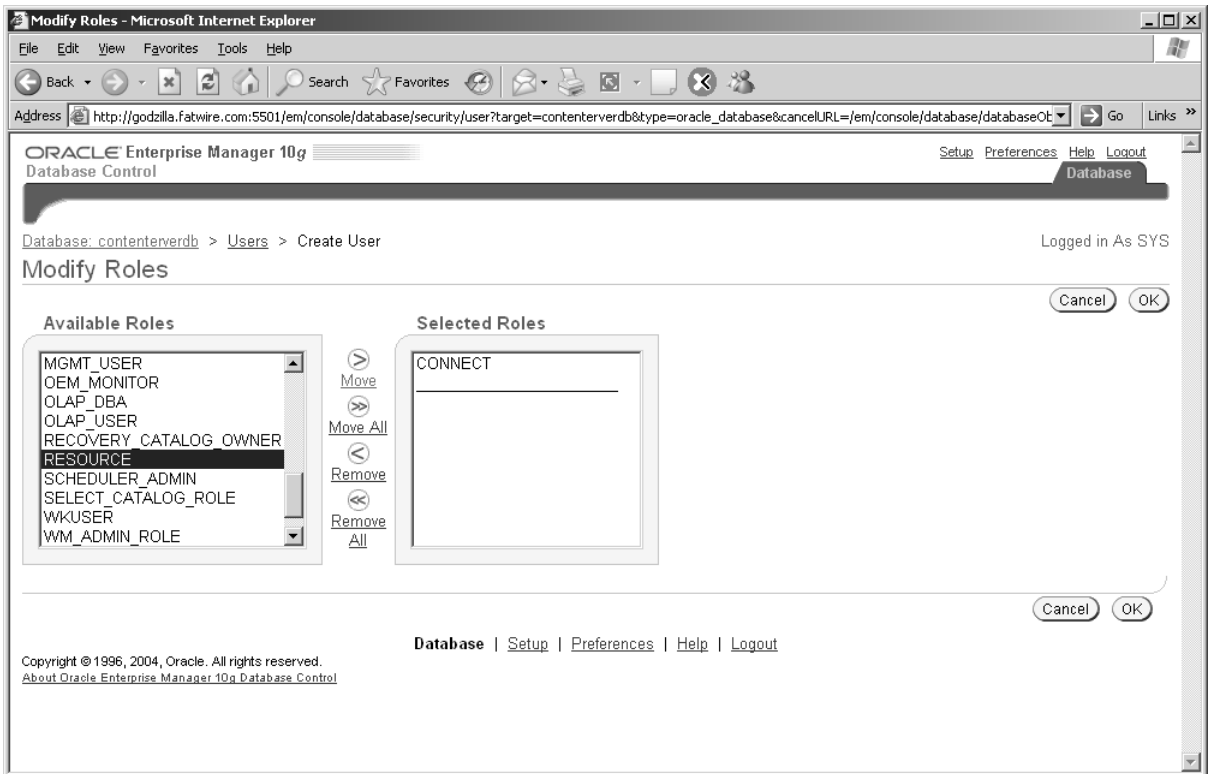


5. Click the **Roles** tab.

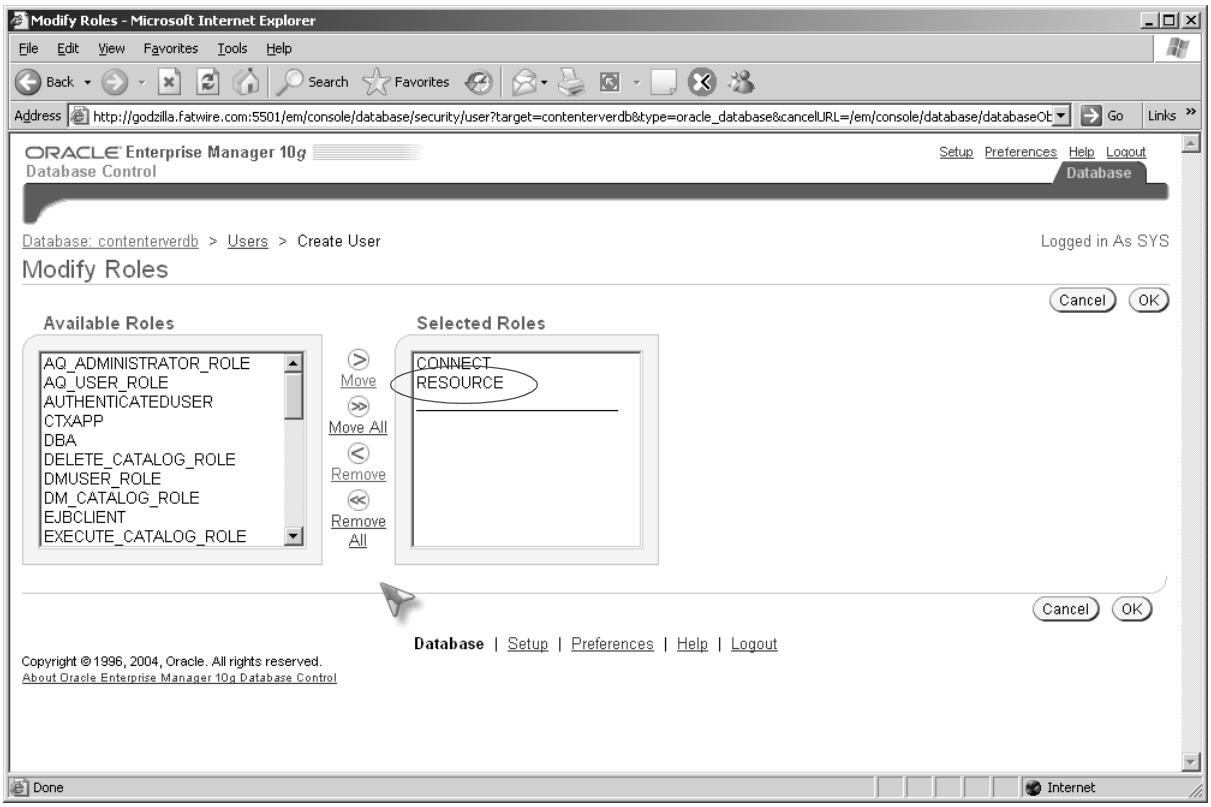
a. Click the **Modify** button.



b. From the list of "Available Roles" (left side), select **Resource** and click the **Move** button.

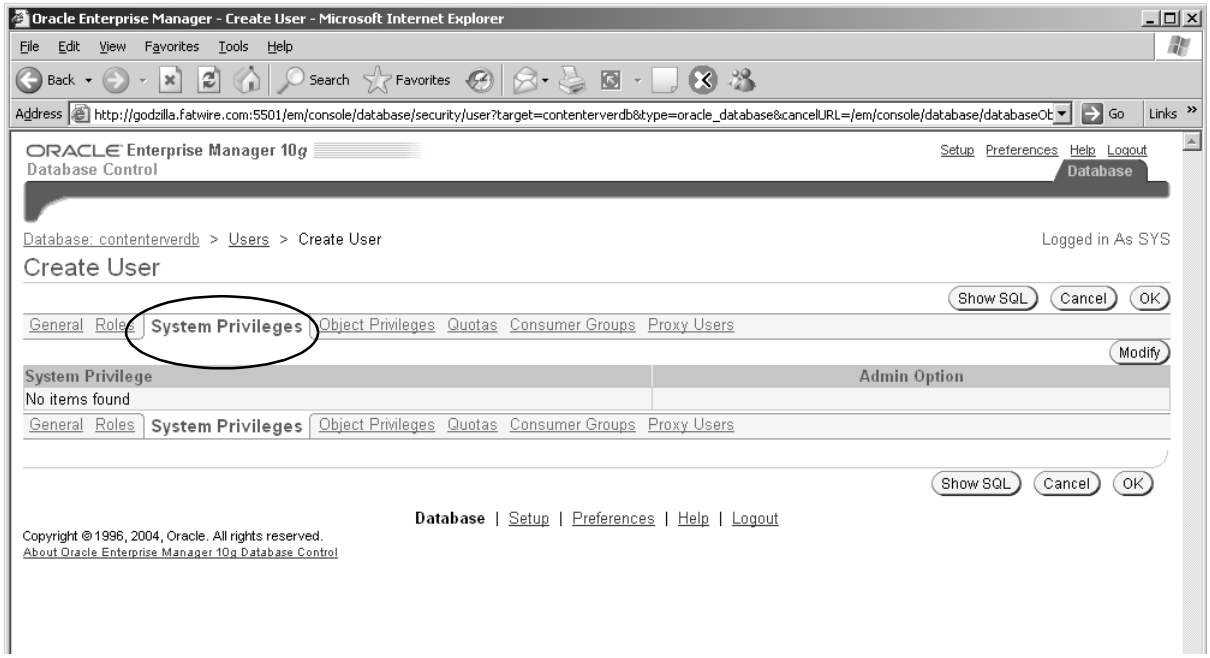


Resource is moved to the “Selected Roles” list.

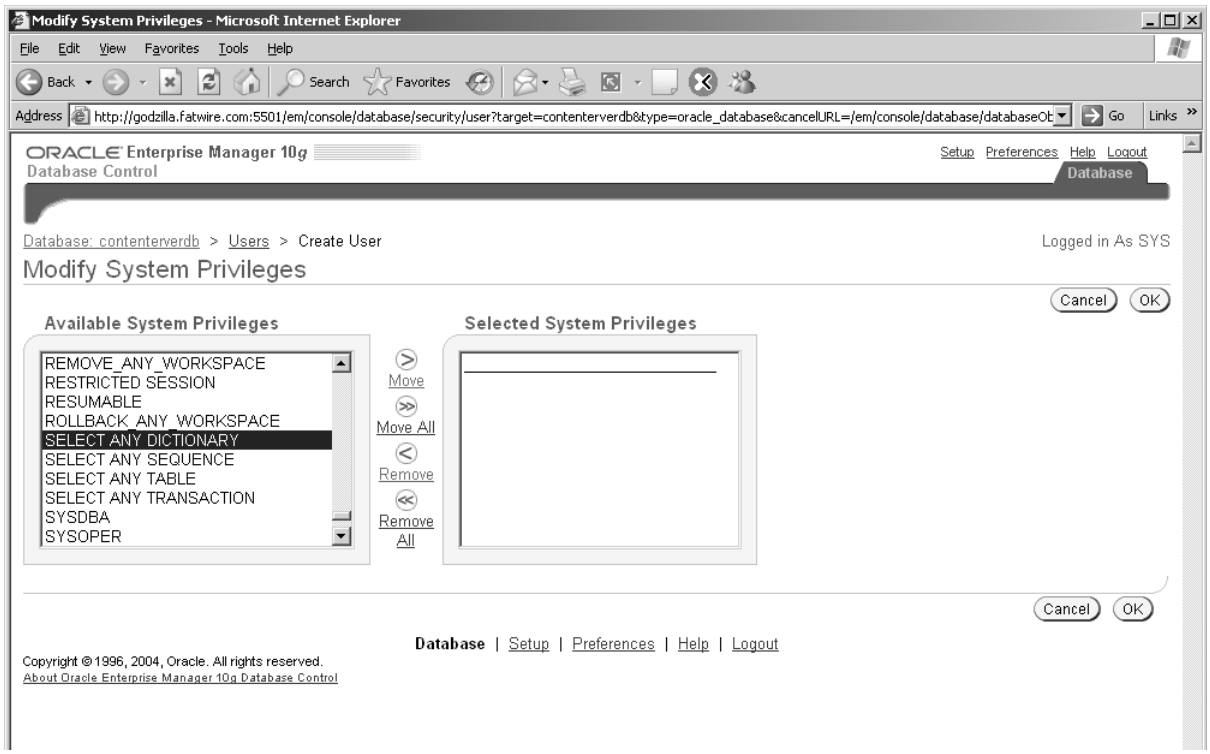


c. Click **OK**.

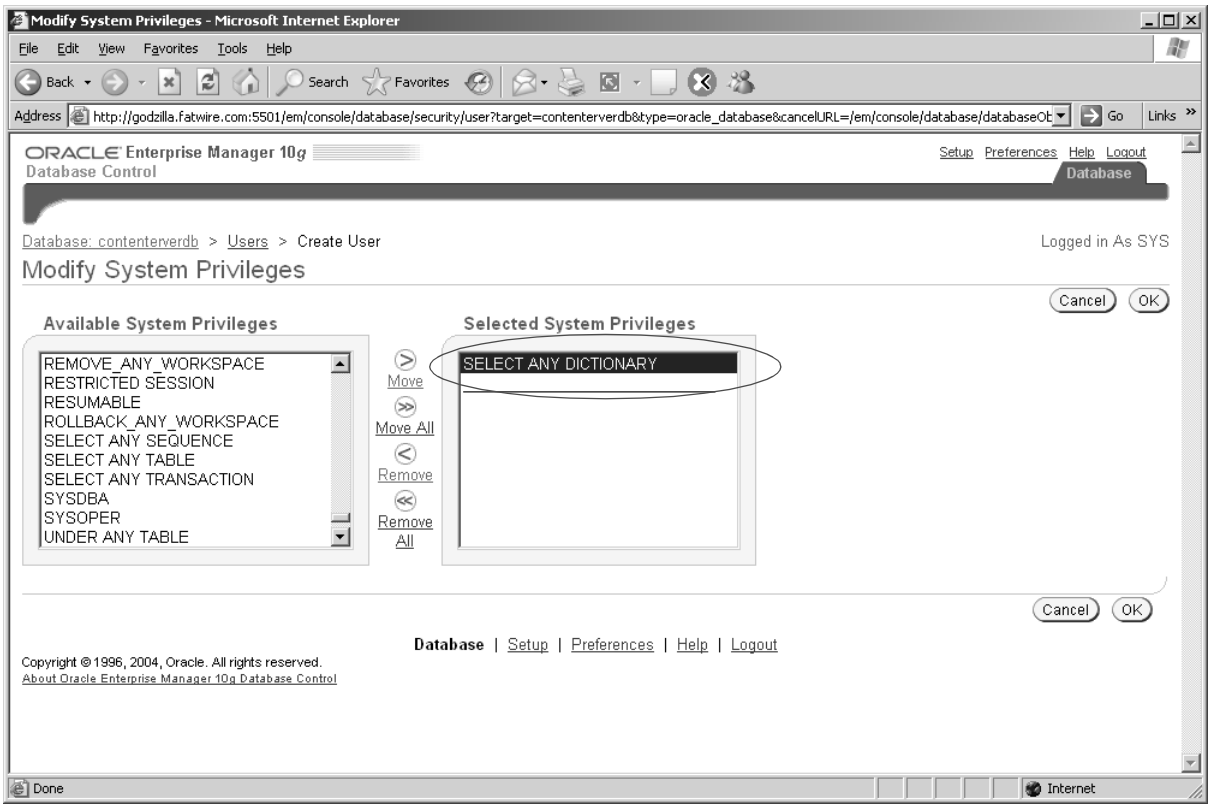
6. Click the **System Privileges** tab.
   a. Click the **Modify** button.



   b. From the list of "Available System Privileges" (left side), choose **Select Any Dictionary** and click the **Move** button.



Select Any Dictionary is moved to the "Selected System Privileges" list.



c. Click **OK**.

The database is now ready for Content Server (Spark).

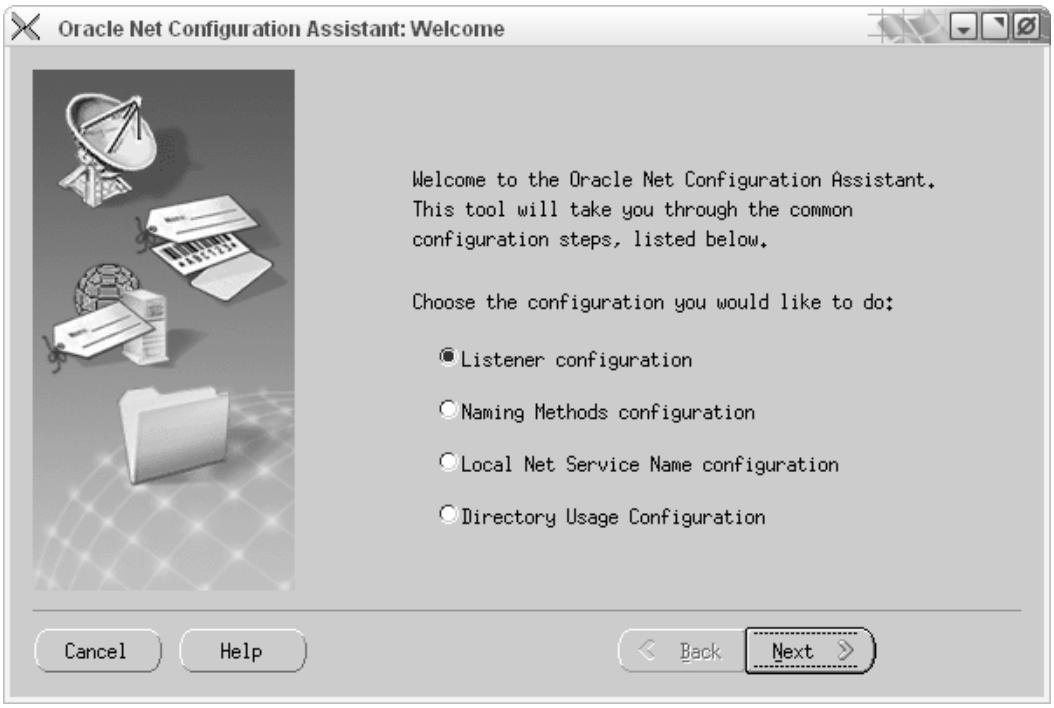
7. In the upper right-hand corner, click **Logout**.
8. If you are using a Type 2 driver, go to "Step III. (For Type 2 Drivers) Install and Configure an Oracle Client," on page 43. Otherwise, the database is ready for Content Server (Spark). You can now create and configure the data source.

Step III. (For Type 2 Drivers) Install and Configure an Oracle Client

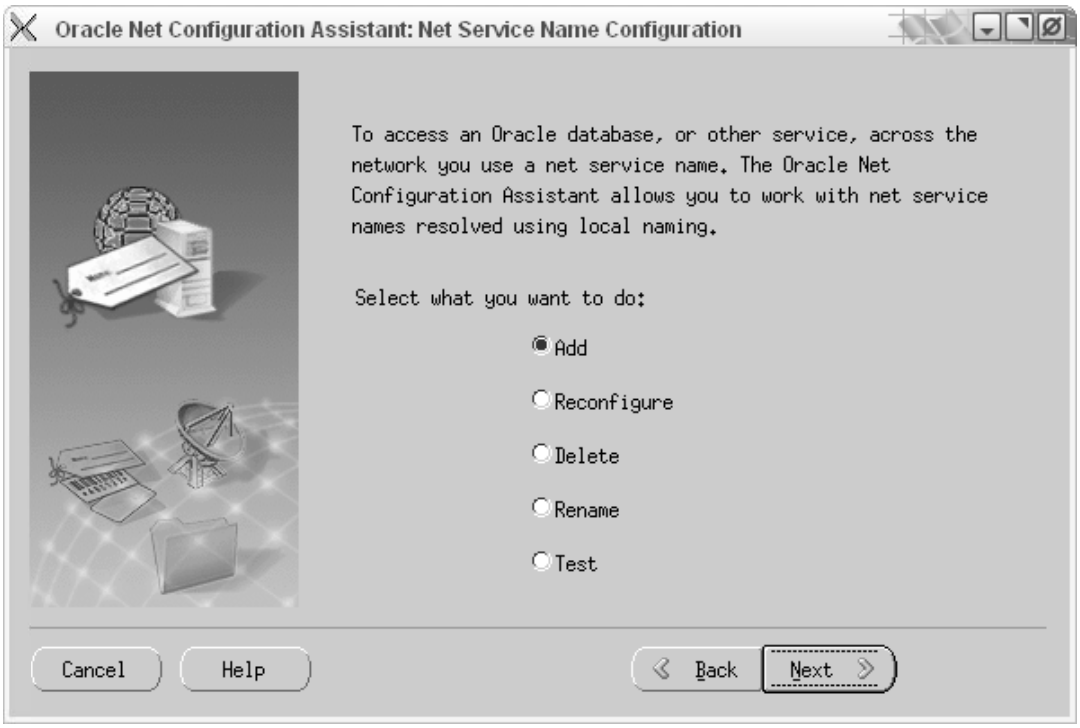
1. Download and install the Oracle AS client.
2. Run the command **`$ORA_HOME/bin/netca`**

> **Note**
>
> This command requires the environment variable `ORA_NAME` to be set and a graphical display to be present in the system to execute successfully.

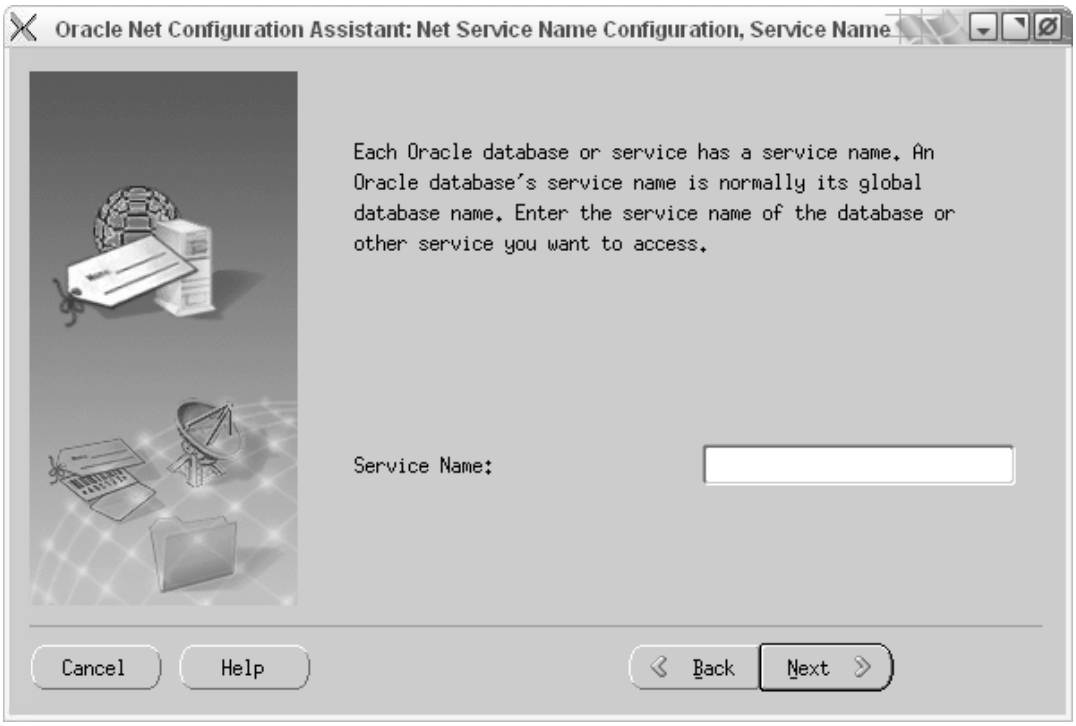
3. Select **Local Net Service Name Configuration** and click **Next**.



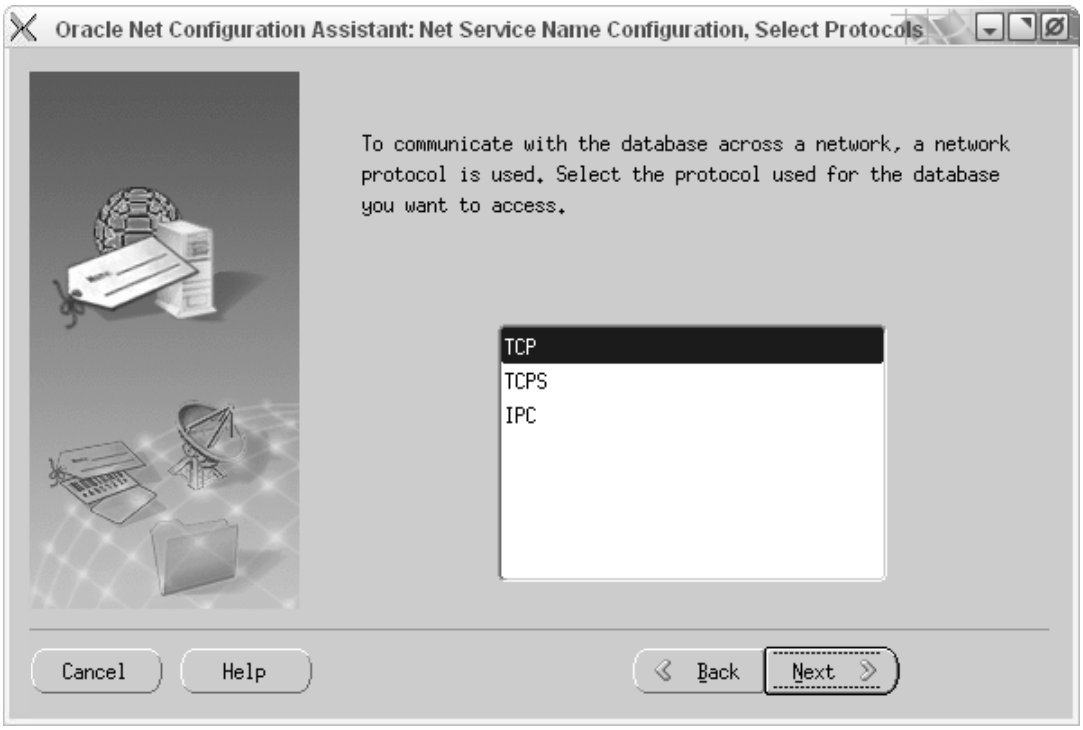
4. Select **Add** and click **Next**.



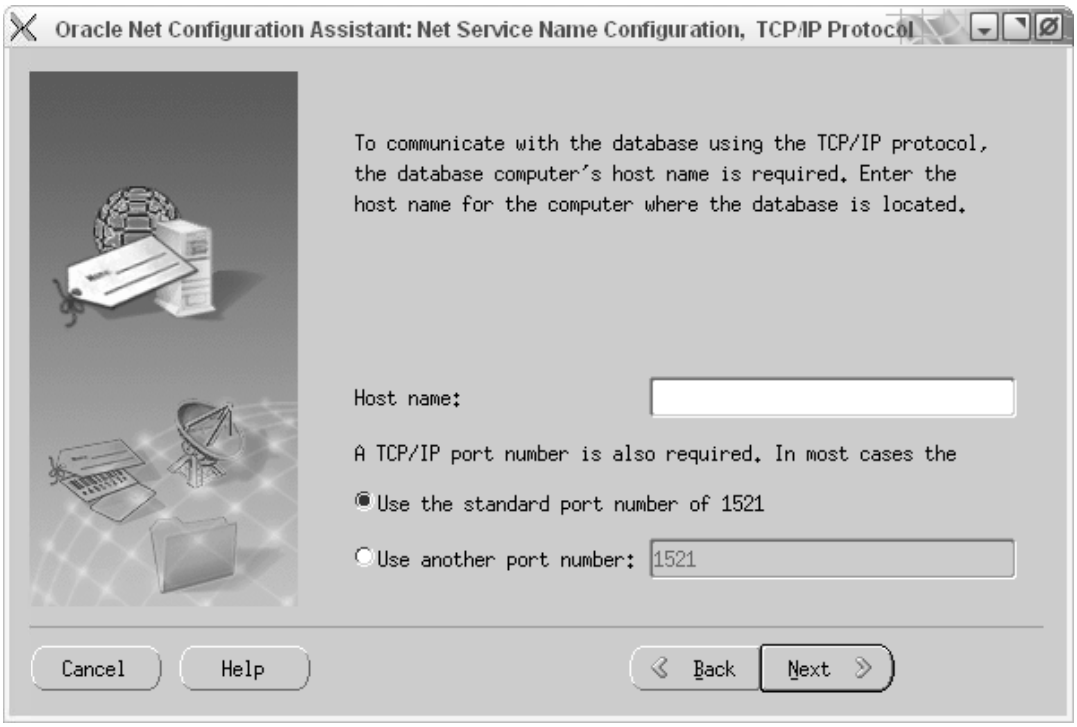
5. Enter a name for this service (this is the SSID of the database on the remote server) and click **Next**.



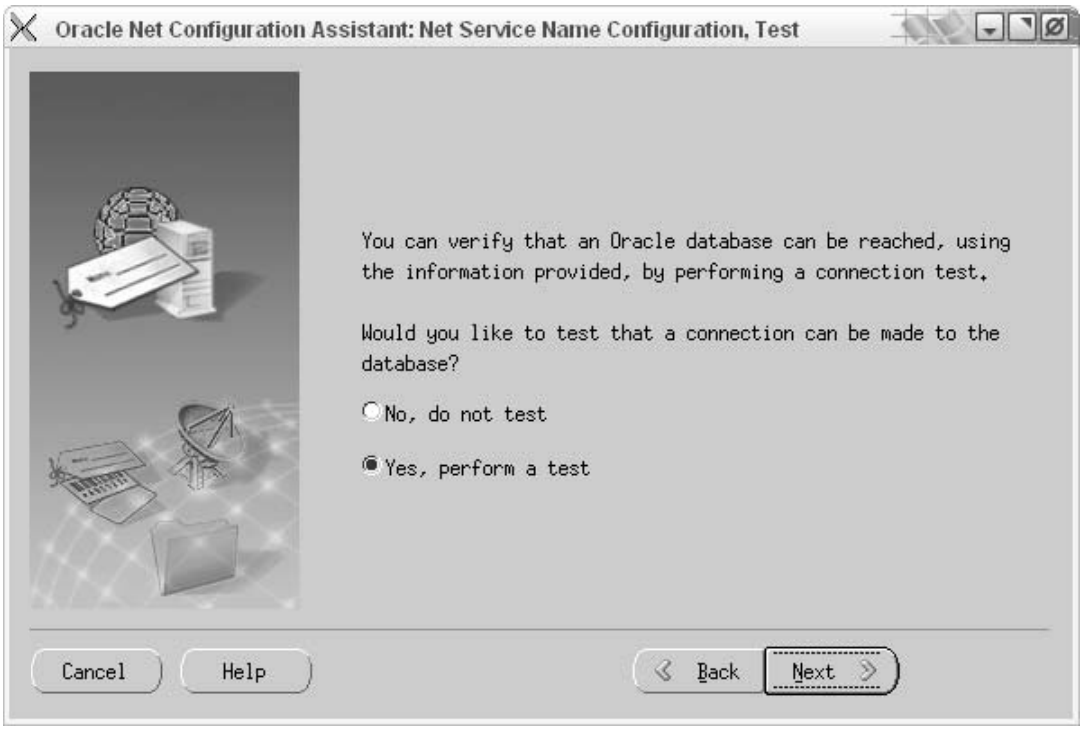
6. Leave the default option of TCP selected and click **Next**.



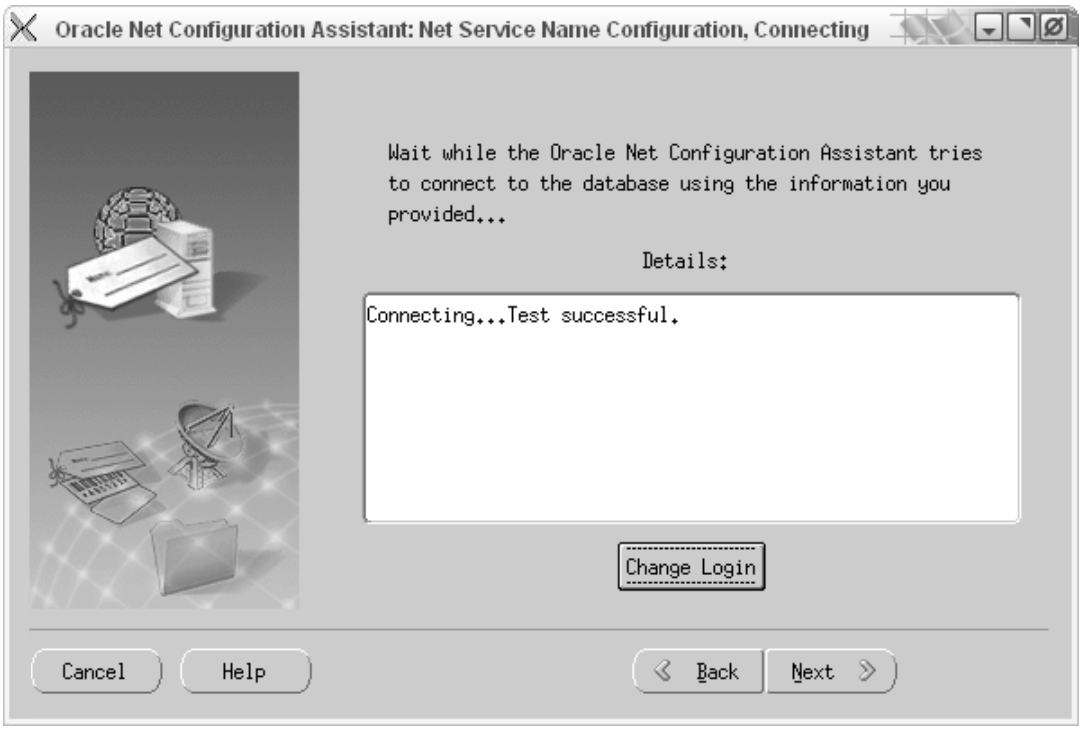
7. Enter the host name (the name of the computer that is hosting the database you are connecting to). If you have instructed Oracle 10*g* to run on a port other than the default, enter the custom port number and click **Next**.



8. Select **Yes, perform a test** and click **Next**.

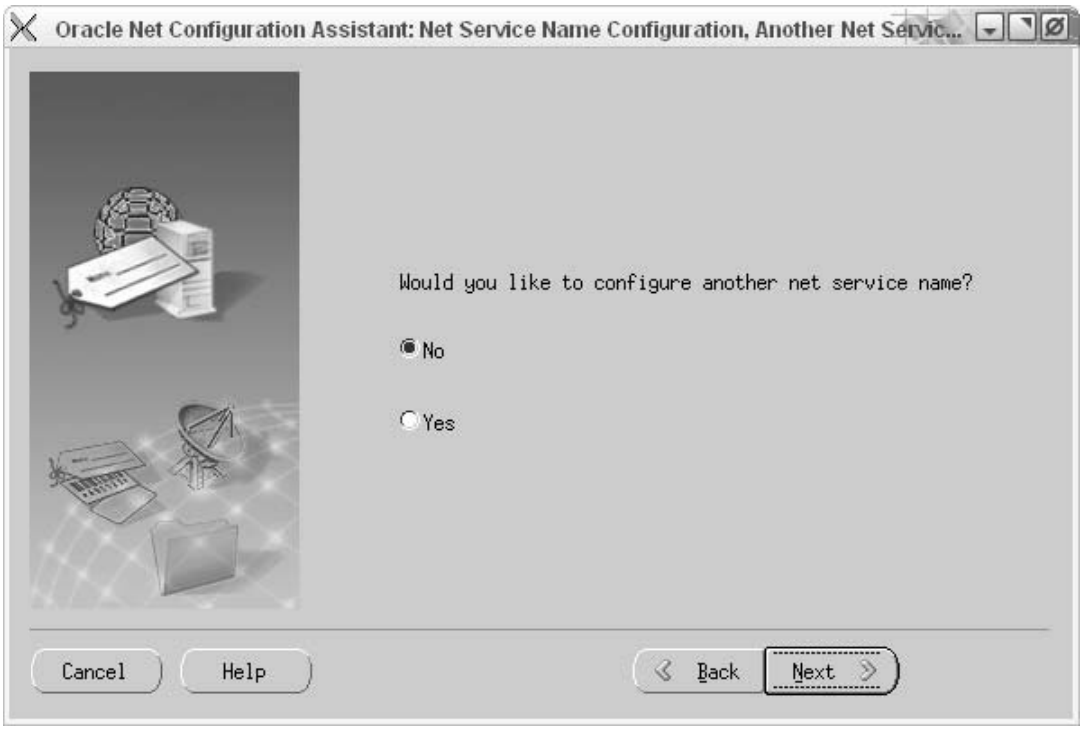


9. The first test will fail. When it does, click on the **Change Login** button, enter the username and password created when the database was constructed, and click **OK**. You will then see the dialog box shown below. Click **Next** to proceed.

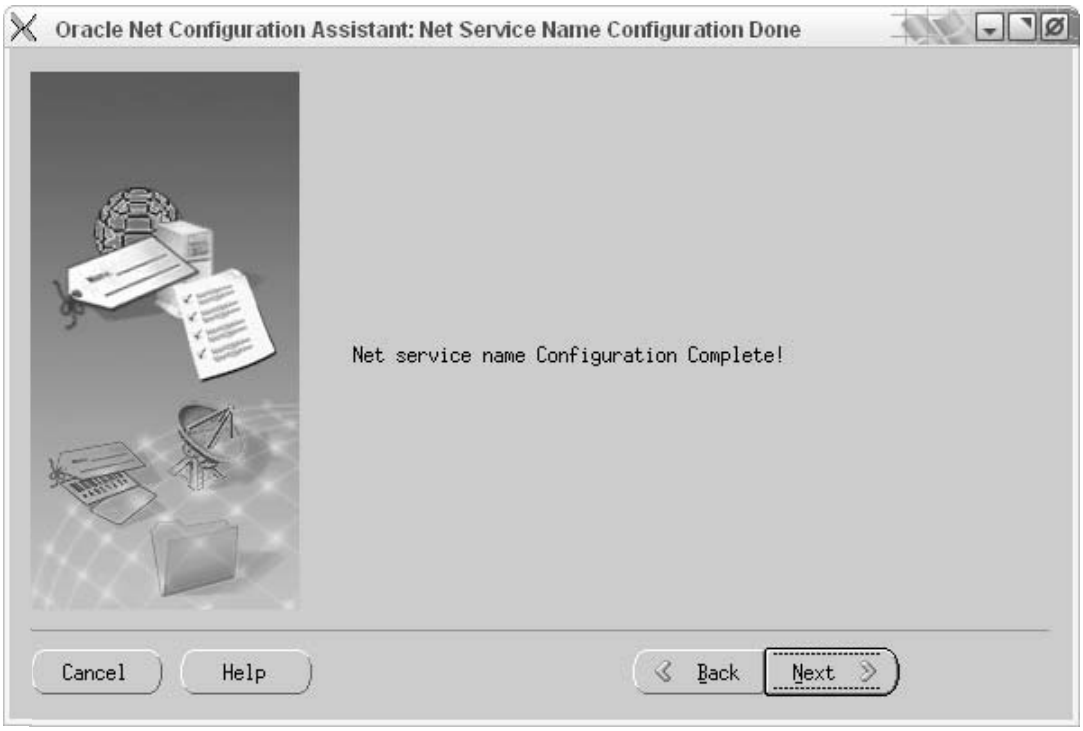


10. Enter a local service name (we suggest the name of the database you will be connecting to) and click **Next**.

11. Leave the default choice of **No** selected and click **Next**.



12. Click **Next** to exit the wizard.



13. Database configuration is complete. You are now ready to create and configure the data source. For instructions, refer to your Content Server (Spark) installation guide.

Chapter 3

Creating and Configuring an MS SQL Server Database

Use this chapter to set up a SQL Server database for your Content Server (Spark) installation. For background information regarding database configuration and users' permissions, see Part 1, "Creating and Configuring a Database."

To create and configure an MS SQL Server database

Note

This section applies to MS SQL Server 2000 SP3 and SP4.

1. Create the database login:
   a. Open "Enterprise Manager."
   b. In the left-hand tree, select **Microsoft SQL Servers** > **SQL Server Group** > **(Local)** > **Security**.
   c. Right-click on **Logins** and select **New Login...**
      1) Create a user (such as `csuser`), and select the proper authentication method.
      2) Save this user.
2. Create the database:
   a. In the left-hand tree, select **Microsoft SQL Servers** > **SQL Server Group** > **(Local)** > **databases**.
   b. Right-click on **databases** and select **New Database…**
      1) Enter a name (such as `csDB`), then modify the other fields as needed for your installation.
      2) Finish creating this database.
3. Assign account privileges:
   a. Select the newly created database in the left-hand tree and click **Open**.
   b. Right-click on **Users** and select **Add new Database User…**
   c. In the drop-down list, choose the user created in step c of this procedure. In the **Permit role membership** list, check the box next to **db_owner**.
   d. Save the new user.
4. Database configuration is complete. You are now ready to create and configure the data source. For instructions, refer to your Content Server (Spark) installation guide.

Chapter 4

Creating and Configuring an IBM DB2 Database

Use this chapter to set up an IBM DB2 database for your Content Server (Spark) installation. For background information regarding database configuration and users' permissions, see Part 1, "Creating and Configuring a Database."

To create and configure a DB2 database for Content Server (Spark)

Note

This section applies to **DB2 version 8.*x***.

1. Open DB Control Center (**`db2cc`**).
2. Browse to the instance under which you want to create the new database.

   If you do not have an existing instance in the left-hand tree, do the following:

   a. Right-click **Instances** and click **Add…**

   b. Fill in the form provided (or click **Discover**) then click **OK**.
3. Right-click **Branch Databases > Create > Database Using Wizard…**
4. In the "Create Database Wizard," fill in the following screens as indicated:

   a. "Database name"
   Enter a unique database name (such as `CSDB2`), then click **Next**.

   b. "Specify how and where to store the user tables."
   Leave the default option **Low maintenance** selected and click **Next**.

   c. "Specify how and where to store the system catalog tables."
   Leave the default option **Low maintenance** selected and click **Next**.

   d. "Specify how and where to store system temporary tables."
   Leave the default option **Low maintenance** selected and click **Next**.

   e. "Tune the performance of this database." Click **Next**.

   f. "Specify the locale for this database."
   Complete the following steps:

      1) In the **Code Set** drop-down list, select **UTF-8**.

      2) Under **Collating Sequence**, leave the default option selected.

      3) Click **Next**.

   g. Review the actions that will take place when you click **Finish**, then click **Finish**.
5. A DB2 message box appears, giving you the option to run the "Configuration Advisor." Click **No**.

   A new database (with the name you provided in step 4) is now available in the left-hand tree.
6. In the left-hand tree, right-click **Buffer Pools > Create**.
7. In the "Create Buffer Pool" dialog box, do the following:

   a. In the "Buffer Pool name" field, add a unique name (such as `CSBUFFER32`).

   b. In the **Page size** drop-down list, select **32**.

   c. Click **OK**.
8. In the left-hand tree, right-click **Table Spaces > Create**.

9. In the "Create Table Space Wizard," fill in the following screens as explained below:
   a. "Specify a name for your table space."
   Enter a unique name (such as `csTableSpace`) in the "Table Space name" field. Then click **Next**.
   b. "Specify the type of table space you want to create."
   Leave the default value and click **Next**.
   c. "Specify a buffer pool for your new table space."
   Select the buffer pool created in step 7 of this procedure and click **Next**.
   d. "Select the space management system that you want to use."
   Leave the default option **System-managed space (low maintenance)** selected and click **Next**.
   e. "Define containers for this table space."
   Click **Add**, then complete the following steps:
      1) In the "Define Container" dialog box, enter a unique name for this container (such as `CScontainer`).
      2) Under "Current Directory," select a location for this table space (note that you must select a physical location on a mounted disk where you want to place this table space; if you do not have an acceptable location at this point you should create one). Once you have selected a location, click **OK**.
      3) Click **Next** in the "Define Container" dialog box.
   f. "Specify the extent and prefetch sizes for this table space."
   Leave the default options selected and click **Next**.
   g. "Select hard drive specifications."
   Select the appropriate option for your physical media type from the list and click **Next**.
   h. "Specify the dropped table recovery option for your new table space." Click **Next**.
   i. Review the actions that will take place when you click **Finish**, then click **Finish**.
10. Repeat step 9 of this procedure to create a temporary table space, making the following adjustments to the procedure:
   a. When completing step 9a, indicate in the name that this is a temporary table space.
   b. When completing step 9b, select **System Temporary** for the type of table space.
11. In the left-hand tree, select **User and GroupObjects** and right-click **DB Users > Add**.
   a. In the "Database" tab, do the following:
      1) Select a user from the **User** drop-down list.

> **Note**
>
> The drop-down list contains all valid system users. If there are no valid system users, you must create one before continuing.

2) Under “Grant authorities for the Selected User,” select all of the options. .

> **Note**
>
> This is not recommended for a delivery system. Choose the options that are appropriate for your delivery system)

b. Click the **Table Space** tab and do the following:

1) Click **Add Tablespace**. In the “Add Tablespace” dialog box, select the tablespace created in step 9 of this procedure and click **OK**.

2) In the “Table Space” tab, the new table space is now selected, but has a ∅ symbol next to it. Select **Grant** from the **Privileges** drop-down list (located near the bottom of the tab).

c. Repeat step b for the temporary table space created in step 10.

d. Optionally, repeat step b to add the default table space `USERSPACE1`.

> **Note**
>
> The default table space was created with the database. Therefore its location is not under your control.

e. Click **OK**.

12. In the left-hand tree, right-click the database created in step 4 of this procedure and click **Configure Parameters**. In the list that opens, make the following changes:

a. Change `LOCKLIST/100` to `LOCKLIST/1024`

b. Change `LOCKTIMEOUT/None` to `LOCKTIMEOUT/30`

c. Change `APPLHEAPSZ/256` to `APPLHEAPSZ/1024`

13. Database configuration is complete. You are now ready to create and configure the data source. For instructions, refer to your Content Server (Spark) installation guide.

Chapter 5

Creating and Configuring SQL Anywhere

This appendix shows you how to create and configure an SQL Anywhere database for your Spark installation.

This appendix contains the following sections:

- Step I. Prepare the Environment
- Step II. Create a Database
- Step III. Create a New User for Spark
- Step IV. Next Step

Step I. Prepare the Environment

Before you begin, source the `asa_config.*` file based on your shell. For example:

```
# bash
# cd /opt/sqlanywhere/bin
# source asa_config.sh
```

Step II. Create a Database

1. If SQL Anywhere is not installed, use SQL Anywhere 9.0 NON-OEM software to install it. Run `setup.sh` and follow the on-screen instructions.

   Note

   - Make sure you choose cpu-based license. Choose all CPUs on your machine.
   - We suggest using `/opt/sqlanywhere` as the installation directory.

2. Go to the following directory: `/opt/sqlanywhere/shared/sybcentral42`
3. Run **`scjview`**.
4. Click **Adaptive Server Anywhere** in the tree and select the **Utilities** tab.
5. Run the “Create Database Wizard,” selecting the following options as you proceed:
   - Leave the checkbox selected for jConnect meta-information support. You must have jConnect meta-information support.
   - Do not select the **Encrypt the database** checkbox.
   - Select the check boxes for **Ignore trailing blanks**, **Case sensitivity**, and **Create SYSCOLUMNS and SYSINDEXES**.
   - Accept the default page size of `2048`.
   - Accept the default collation.
   - Choose to “connect to database." Choose a server name (usually this is the machine name) and give the database a name (such as `sparkdb`).
   - Click **Finish** to complete database creation.
6. After the “Database Creation Wizard” returns, the database and its objects will be displayed in the tree on the left side.
   a. Right-click on your database and select **Options**.
   b. Set the "Non_keywords" option to **PUBLICATION**. This explicitly turns off the **PUBLICATION** keyword in your database.

Step III. Create a New User for Spark

1. Create a user (named `csuser`) in the current database by right-clicking **Users and Groups** and following the directions in the wizard. (Make sure to grant the user the required “Resource” permissions.)

   Note

   Do not give DBA permissions to this user.

2. Test the user you created by completing the following tasks:
   a. Go to the following directory: `/opt/sqlanywhere/bin`
   b. Run **`dbisql`**.
   c. Connect to the database using the newly created user.
   d. Confirm that **PUBLICATION** is turned off.
   e. Do the following queries:

      CREATE TABLE PUBLICATION (id varchar(2))

      DROP TABLE PUBLICATION

      If both the queries do not succeed, there is a problem with your installation. Begin the installation again.
3. Disconnect from the database in SybCentral by completing the following steps:
   a. Right-click on your database in the tree and choose **Disconnect**.
   b. Exit from Sybase Central.
   c. Run **`ps -ef | grep dbeng`**. If you notice dbeng running, your server is still alive. If it is not running, then issue the following command:

      `dbsrv9 /<fullpath>/<database name>.db`
   d. Run **`dbisql`** again to confirm that the server is running.

Step IV. Next Step

Create a new JDBC connection pool and JDBC resource. For instructions, refer to your Content Server (Spark) installation guide.

Part 2

Installing a Web Server

This part describes how to install a web server. It contains the following chapters:

- Chapter 7, “Installing IIS on Windows”
- Chapter 8, “Installing Apache on Solaris or Linux”

Chapter 6

Worksheets for Documenting the Web Server Installation

This chapter contains worksheets listing the web server parameters that you need to track.

Print this chapter. Then, as you install software, fill in the blank fields in these worksheets with the values of the specified parameters. You will save considerable time by doing this. Additionally, if something fails during the installation, the information in these worksheets will be valuable while you are troubleshooting. Use a separate set of worksheets for each installation so that each installation is fully documented.

The worksheets are constructed as tables that are divided into the following categories:

- Key to Sample Values
- Web Server Parameters

Key to Sample Values

The installation worksheets list parameters along with their sample values. Each sample value is classified as one of the following:

- **Default**: the value is automatically created at the time of the installation.
- **Normal**: the value represents the normal configuration for a simple installation. Do not use a different value unless your system requires it.
- **Option**: the value must be chosen from a preset list of options.
- **Suggested**: the value is recommended for the parameter.

> **Note**
>
> A **Suggested** account name has an Example password value. We strongly recommend that you select a password for this account that is appropriate for the security of your system.

- **Example**: the value is only an example that must be replaced by the value that is appropriate for your installation. The example value is not likely to be valid in your environment.

Web Server Parameters

Table 1: IIS Web Server Parameters

Parameter	Shown As	Comments	Your Value
Web Version	*WebVersion*	Example: Apache 1.3.37	
Web Host Name	*WebHost*	Example: `jeeves`	
Web Host IP Address	*WebIP*	Example: `104.222.111.155`	
Web Server Port	*WebPort*	Default: `80`	
IIS Only: Filter Name (ISAPI plug-in name)	*FilterName*	Suggested: `iisforwardfilter`	
Apache Only: Apache Root Directory	*ApacheRoot*	Example: `/usr/apache`	

Table 2: Apache Web Server Parameters

Parameter	Shown As	Comments	Your Value
Web Version	*WebVersion*	Example: Apache 1.3.37	
Web Host Name	*WebHost*	Example: `jeeves`	
Web Host IP Address	*WebIP*	Example: `104.222.111.155`	
Web Server Port	*WebPort*	Default: `80`	
IIS Only: Filter Name (ISAPI plug-in name)	*FilterName*	Suggested: `iisforwardfilter`	
Apache Only: Apache Root Directory	*ApacheRoot*	Example: `/usr/apache`	

Chapter 7

Installing IIS on Windows

This chapter explains how to install and test Microsoft's Internet Information Services (IIS). It contains the following sections:

- Step I. Install IIS
- Step II. Document Your IIS Installation
- Step III. Verify the Installation
- Next Step

Note

Typically, IIS is either partially or fully installed on most Windows 2000 machines.

- If IIS is only partially installed or not installed, start with the first section, "Step I. Install IIS," on page 66.
- If IIS is fully installed, start with the section "Step II. Document Your IIS Installation," on page 66.

Step I. Install IIS

If IIS is not installed or is only partially installed, follow Microsoft's instruction for installing IIS on a Windows 2000 system.

As a convenience, here is a quick synopsis of the instructions:

1. Select **Start** > **Settings** > **Control Panel**.
2. Select **Add/Remove Programs**.
3. Select the **Add/Remove Windows Components** tab on the left.

   The **Add/Remove Windows Components Wizard** appears.
4. Select **Internet Information Services (IIS)** and then follow the instructions for installing it.

Step II. Document Your IIS Installation

We strongly recommend that you document the details of your IIS installation in Table 1, "IIS Parameters."

Table 1: IIS Parameters

Parameter	What It Holds	Your Value
Web Version (*`WebVersion`*)	The version number of the IIS software that you installed.	
Web Host Name (*`WebHost`*)	The name by which the installation machine is known on the network.	
Web Host IP Address (*`WebIP`*)	The numeric Internet Protocol address assigned to the web server host machine.	
Web Server Port (*`WebPort`*)	The port number assigned for web server communications. By default, it has the value `80`.	

Step III. Verify the Installation

After you have installed IIS, you start it and then browse to it in a web browser to determine whether it is serving pages as it should.

A. Start IIS

You can start the various IIS services in various ways. To be sure that all the necessary services are running, start IIS from the **Services** node.

To start IIS services

1. Right-click on the **My Computer** icon.

2. Select **Manage** from the right-mouse menu.
3. In the **Computer Management** dialog box, expand the **Services and Applications** node in the tree.
4. Select **Services**.
5. In the list of services on the right, right click **IIS Admin Service**.
6. Select **Start** from the right mouse menu.

To start or stop the default web site only

1. Right-click on the **MyComputer** icon.
2. Select **Manage** from the right mouse menu.
3. In the **Computer Management** window, expand the **Services and Applications**. node in the tree.
4. Expand the **Internet Information Services** node.
5. Right-click on **Default Web Site**.
6. Select **Start** or **Stop**, as appropriate, from the right mouse menu.

B. Verify that IIS is Serving Pages

To verify that IIS can serve pages, test it from both the server that is hosting it and from another browser on the network.

To verify that IIS can serve pages

1. Start a browser on the host on which IIS is running.
2. From the browser, go to the following URL:

   `http://WebHost:WebPort`
3. Do one of the following:
   - If the browser displays the IIS home page, then IIS is installed and running properly. Continue to step 4.
   - If the browser returns an error, consult Microsoft's documentation, determine what went wrong, and fix it before you continue.
4. Start a browser on another machine on your network (a host other than the machine hosting IIS).
5. From the browser, go to the following URL:

   `http://WebHost:WebPort`

   If the browser displays the IIS "Under Construction" page, then IIS is installed and running and the network naming service appears to be working properly.

Next Step

Configure the web server to run with WebLogic and Content Server (Spark). For instructions, refer to the installation guide for your configuration.

Chapter 8

Installing Apache on Solaris or Linux

This chapter describes how to install and configure Apache HTTP Server. As previously mentioned, you can install Apache on the same machine that will host WebLogic and Content Server (Spark) or you can install and use it on a separate host.

This chapter contains the following sections:

Step I. Install Apache

1. Apache HTTP Server can be pre-installed on Solaris 8, Solaris 9, Linux RedHat, and Linux SuSE systems. Determine whether Apache is installed on the environment(s) on which you plan to run it.
2. Do one of the following:
   - If Apache is already installed, continue with "Step II. Document Your Apache Parameters," on page 70.
   - If Apache is not already installed, you can do one of the following:

- Install it from your source medium.
- Download it from the Internet.
- Build it from source; that is, select the modules and compile the Apache executable yourself. If you want to build it from source, refer to the information that the Apache Foundation makes available at `http://www.apache.org/` and follow their instructions.

Step II. Document Your Apache Parameters

We strongly recommend that you document the details of your Apache installation in Table 2, "Apache Parameters."

Table 2: Apache Parameters

Parameter	What it Holds	Your Value
Web Server Version (*`WebVersion`*)	The version of Apache that the host is running. Note that you must use a version that Content Server (Spark) supports.	
Web Host Name (*`WebHost`*)	The name by which the Apache host machine is known on the network.	
Web Host IP Address (*`WebIP`*)	The numeric Internet Protocol address assigned to the Apache host machine.	
Web Server Port (*`WebPort`*)	The port number assigned for Apache communications. By default, it has the value 80.	
Apache Root Directory (*`ApacheRoot`*)	The top-level directory in which Apache is installed. Immediate subdirectories of *`ApacheRoot`* include `bin` and `conf`.	

Step III. Verify that Apache Contains the Correct Module

Note

This section applies only to Apache version 1.3x.

Apache is modular software, built from a set of modules. WebLogic Server requires that the `mod_so.c` module be present on the machine that is hosting the Apache web server. Please verify that your Apache server contains this module by using the command `httpd` with the `-l` option and search for `mod_so` in the output.

For example:

```
$ ApacheRoot/bin/httpd -l | grep 'mod_so'
mod_so.c
```

Examine the output and do one of the following:

- If the output from the preceding command contains `mod_so.c`, then your version of Apache contains the correct module. Proceed to "Step IV. Verify that Apache Runs Properly," on page 71.
- If the output from the preceding command does not contain `mod_so.c`, you must rebuild and reinstall Apache. For guidelines, see "Step I. Install Apache," on page 70.

Step IV. Verify that Apache Runs Properly

In this step, you will start Apache and verify that it is running properly. For verification instructions, see the Apache web site (given in "Step I. Install Apache," on page 70).

Next Step

Configure Apache to run with WebLogic and Content Server (Spark). For instructions, refer to the installation guide for your configuration.

Part 3

LDAP Integration

This part describes how to configure an LDAP server for your Content Server web application. It contains the following chapter:

- Chapter 9, “Integrating Content Server with LDAP”

Note

Spark uses a bundled LDAP server. Spark must be automatically integrated with the LDAP server during the Spark installation process. The Spark installer performs the integration as explained in “Automatic Integration with LDAP,” on page 76.

Chapter 9

Integrating Content Server with LDAP

Content Server administrators who need to manage users and their ACLs outside the Content Server database can use an LDAP server to store user names and attributes. Additionally, the LDAP server can be used to manage roles and sites.

Content Server supports the following integration options:

- External LDAP servers, which must be integrated manually after Content Server is installed.
- Bundled LDAP servers, which are packaged with their application servers. Bundled LDAP servers can be integrated either automatically by the CS installer or manually following the CS installation process, depending on whether the installation is a portal or a web application.

This appendix shows you how to integrate LDAP servers with Content Server.

Note

CS-LDAP integration is made possible by the Directory Services API, which also supports the following features:

- The plug-in that functions as Content Server's native user manager. The plug-in authenticates users by means of the native Content Server user management tables: `SystemUsers` and `SystemUserAttrs`
- External user managers, which must be customized in order to authenticate users.

This appendix contains the following sections:

- LDAP Integration Overview
- Automatic Integration with LDAP
- Manual Integration with LDAP
- Post-Integration Management of ACLs, Users, Sites, and Roles
- Reference: Example ldif Files

LDAP Integration Overview

Integrating an LDAP server with a Content Server installation requires you (or the CS installer) to complete the following basic steps:

1. Configure Content Server to communicate with the LDAP server. This step entails configuring properties in Content Server's `futuretense.ini`, `dir.ini`, and `futuretense_xcel.ini` files in order to enable CS-LDAP communication.
2. Configure the LDAP server to:
   a. Recognize the user that Content Server will invoke to access the LDAP server.
   b. Grant that user the correct permissions to the LDAP server.
3. Populate the LDAP server with Content Server's ACLs and users, and optionally, its sites and roles (if you choose this option, you must make sure that the same sites and roles exist in the Content Server database). Additionally, the users must be enabled for the sites.

Whether you integrate manually or automatically depends on whether your Content Server configuration uses bundled or external LDAP, and whether you are installing a portal or a web application. The table below summarizes the integration scenarios.

Table 1: LDAP Integration Methods

Installation Type	LDAP Integration Method	
	Bundled LDAP	**External LDAP**
Portal	Automatic integration[a]	—
Web	Automatic integration[a] Manual integration[b]	Manual integration[b]

a. Automatic integration is performed by the CS installer during the Content Server installation process. For integration information, see "Automatic Integration with LDAP."

b. Manual integration is performed by the installation engineer after Content Server is installed. For integration procedures, see "Manual Integration with LDAP," on page 77.

Automatic Integration with LDAP

During the Content Server installation process, you have the option to automatically integrate with LDAP. If you select this option (based on Table 1), the integration steps above will be performed by the CS installer. That is:

1. The CS installer configures Content Server to communicate with the LDAP server as explained in step 1, above.
2. If the installation is to run on WebSphere, the CS installer configures the LDAP server as explained in step 2, above. For all other platforms, the LDAP administrator must configure the server.

3. The CS installer populates the LDAP server with the following information (as explained in step 3, above):
   - Content Server's default ACLs and roles.
   - Sample sites that you might have chosen to install in Content Server, and users of the sample sites. The CS installer also enables the users for the sample sites.

Manual Integration with LDAP

Note

For portal installations, manual integration is not an option. If an existing installation is not LDAP-integrated, it must be re-installed with the LDAP integration option enabled. However, once integration is complete, management operations on ACLs, users, sites, roles, and property files are performed manually (instructions are given in "Post-Integration Management of ACLs, Users, Sites, and Roles," on page 88).

If your CS system is to use external LDAP, you must integrate the LDAP server *manually*, regardless of whether you are creating a new installation or modifying an existing installation to use LDAP. Below are the basic steps you will complete. The steps apply to both new and existing installations (all of which are web-based, as the current release does not support external LDAP for portals):

Make Sure That Pre-Requisite Systems Are Installed:

1. Make sure that the LDAP server is installed. For more detailed information, see "Step 1. Make Sure the LDAP Server is Installed," on page 78.
2. Skip this step if you are integrating an existing system with external LDAP. Otherwise, make sure that Content Server is installed *and tested*. For more detailed information, see "Step 2. Make Sure That Content Server is Installed and Tested," on page 78.

Configure Content Server and the LDAP Server to Communicate:

3. Configure the Content Server properties that enable Content Server to communicate with the LDAP server. For detailed instructions, see "Step 3. Configure Content Server Properties," on page 79.
4. Configure the LDAP server to recognize the user that Content Server will use to access the LDAP server and to grant that user the correct permissions to the LDAP server. For detailed instructions, see "Step 4. Configure the LDAP Server," on page 83.

Populate the LDAP Server with Content Server's ACLs, Users, Sites, and Roles:

5. Ensure that the LDAP server stores a mail attribute (e-mail address) for your users. For detailed instructions, see "Step 5. Check the mail Attribute," on page 83.

6. Use the LDAP interface to create LDAP user groups whose names exactly match the Content Server system ACLs. For installations using web services, include the corresponding ACLs. For detailed instructions, see “Step 6. Create LDAP User Groups (Content Server ACLs),” on page 83.
7. Use the LDAP interface to create users and to assign users to the groups. For detailed instructions, see “Step 7. Create Required Users and Their Group Memberships,” on page 84.
8. (Optional) If you plan to use LDAP attribute mapping for sites and roles, create the appropriate sites and roles in LDAP. For detailed instructions, see “Step 8. (Optional) Create Sites and Roles in the LDAP Server,” on page 85.

If You Completed Step 8:

9. Make sure that the sites and roles you created in step 8 also exist in the CS database. For detailed instructions, see “Step 9. If You Completed Step 8,” on page 87.

Test the Configuration:

10. Test the configuration. For detailed instructions, see “Step 10. Test the LDAP Integration,” on page 87.

Resume Your Previous Operations:

11. Once LDAP integration passes the test, you can resume your operations. If they involve managing ACLs, users, sites, and roles in the LDAP-integrated system, see “Post-Integration Management of ACLs, Users, Sites, and Roles,” on page 88 for guidelines.

Note

The steps below are detailed versions of the steps that are given on page 77.

Step 1. Make Sure the LDAP Server is Installed

Content Server can be integrated with LDAP servers that support the LDAP-2 protocol or any other protocol (although, in the latter case, FatWire does not support write operations from the Content Server interface to the LDAP server). The currently supported LDAP servers are listed at the following URL:

`http://www.fatwire.com/Support/cs`*`xxx`*`.html`

where *xxx* represents the Content Server version number (for example, 630). Check that one of the listed LDAP servers is installed.

Step 2. Make Sure That Content Server is Installed and Tested

Before you can begin integrating with LDAP, you need to make sure that Content Server has been installed and tested on a supported configuration. The currently supported configurations are listed at the URL that is given directly above, in Step 1 of this procedure.

Step 3. Configure Content Server Properties

Configure the following property files in order for Content Server to communicate with the LDAP server:

- `futuretense.ini`
- `dir.ini`
- `futuretense_xcel.ini`, if you will be managing sites and roles directly in the LDAP server (unlike ACLs and users, sites and roles need not be managed in the LDAP server).

Set Property Values in futuretense.ini

1. Start the Property Editor and open `futuretense.ini`.
2. Select the **Authentication** tab.
3. Set the following values for the following properties:

Property (In futuretense.ini)	Value
`cs.manageproperty`	`dir.ini`
`cs.manageUser`	`com.openmarket.directory.jndi.auth.JNDILogin`

4. Select **File** > **Save** to save the values.
5. Select **File** > **Close**.
6. Continue to the next procedure.

Set Property Values in the dir.ini File

1. Open `dir.ini` in the Property Editor.
2. Select the **Attribute Names** tab and set the following values for the following properties:

Property (in dir.ini)	Value
`cn`	iPlanet: `cn` Active Directory: `cn`
`loginattribute`	iPlanet: `uid` Active Directory: `cn`
`password`	iPlanet: `userPassword` Active Directory: `password`
`uniquemember`	iPlanet: `uniquemember` Active Directory: `member`

Property (in dir.ini)	Value
`username`	iPlanet: `uid` Active Directory: `sAMAccountName`

3. Select the **Global Data** tab and set the following values for the following properties:

Property (in dir.ini)	Value
`groupparent`	iPlanet: `ou=groups,dc=`*`companyname`*`,dc=com` Active Directory: `cn=groups,dc=`*`companyname`*`,dc=com`
`peopleparent`	iPlanet: `cn=people,dc=`*`companyname`*`,dc=com` Active Directory: `cn=users,dc=`*`companyname`*`,dc=com`

4. Select the **Interface Implementations** tab and specify the following values for the following properties:

Property (in dir.ini)	Value
`className.IDir`	`com.openmarket.directory.jndi.JNDIDir`
`className.IName`	`com.openmarket.directory.jndi.NameWrapper`
`className.IUserDir`	`com.openmarket.directory.jndi.LDAPUserDir`

5. Select the **JNDI SPI Env** tab and specify the following values for the following properties:

Property (in dir.ini)	Value
`jndi.baseurl`	`ldap://`*`servername:port`*
`jndi.connectAsUser`	If Content Server can query the LDAP server for information as the user who is logged in to the Content Server interface and is making the query, set this property to `true`. (The same user must be defined in the LDAP server.) If Content Server must query the LDAP server as a specific user other than the user who is logged in to the Content Server interface, set this property to `false`. Then be sure to specify a valid user name/password combination with the `jndi.login` and `jndi.password` properties.
`jndi.custom`	(leave this value blank)

Property (in dir.ini)	Value
`jndi.login`	If the `jndi.connectAsUser` property is set to `false`, specify the fully qualified, fully distinguished name of the user account that Content Server will use to query the LDAP server. (The same user must be defined in the LDAP server.)
`jndi.password`	If the `jndi.connectAsUser` property is set to `false`, specify the password for the user account that Content Server will use to query the LDAP server. This value is encrypted.

6. Select the **Schema Defaults** tab and specify the following values for the following properties:

Property (in dir.ini)	Value

`defaultGroupAttrs`	iPlanet:
	`objectClass=top &objectClass=groupofuniquenames`
	Active Directory:
	`objectClass=top&objectClass=group`

`defaultPeopleAttrs`	iPlanet:
	`objectClass=person &objectclass=top &objectclass=organizationalperson &objectClass=inetOrgPerson`
	Active Directory
	`objectClass=person &objectclass=top &objectclass=organizationalperson &objectClass=user`

`objectclassGroup`	iPlanet: `groupofuniquenames` Active Directory: `group`
`objectclassPerson`	`person`
`requiredGroupAttrs`	(leave this value blank)
`requiredPeopleAttrs`	Specify all the required user attributes for this LDAP server. For example: `sn=Last Name&cn=Full Name`

7. Select **File** > **Save**.

8. Select **File** > **Close**.

Note

If you need information on properties in the `dir.ini` file that are not listed in the previous tables, see the *Content Server Property Files Reference*.

Set Property Values in the futuretense_xcel.ini File

Note

Skip this procedure if you are not using sites and roles integration in your LDAP server.

1. Start the Property Editor and open `futuretense_xcel.ini`.
2. Select the **Xcelerate** tab.
3. Set the value for the following property:

Property (in futuretense_xcel.ini)	Value
`xcelerate.usermanagerclass`	The value depends on the type of deployment you are using. See the *Property Files Reference* for detailed information about this property. Example value: `com.openmarket.xcelerate.user.LDAP SchemaUserManager`

4. Select the **User Management** tab.
5. Set values for the following properties:

Property (in futuretense_xcel.ini)	Value
`xcelerate.sitesroot`	See the *Property Files Reference* for detailed information about this property. Example value: `ou=sites.dc=<domainname>.dc=com`
`xcelerate.sitenameattr`	See the *Property Files Reference* for detailed information about this property. Example value: `ou`
`xcelerate.displayablenameattr`	The name of the user attribute describing the displayable name, if different from the login name.

6. Select **File** > **Save**.
7. Select **File** > **Close**.
8. Configure the CS system to use the LDAP plug-in by setting the appropriate property values in the `futuretense.ini`, `futuretense_xcel.ini`, and `dir.ini` property files. For detailed instructions, see "Step 3. Configure Content Server Properties," on page 79.

Step 4. Configure the LDAP Server

1. Configure the LDAP server to recognize its user. To do so, enter the values of the properties `jndi.connectAsUser` and `jndi.login` into the LDAP server.
2. Assign the correct permissions to the LDAP user.

Note

Assign permissions judiciously. Once the CS system is configured to communicate with the LDAP plug-in, any CS administrator who connects to the LDAP server as a user with write permissions can still perform management operations on ACLs, users, sites, and roles in the Content Server interface. However, some of the operations will propagate to the LDAP server, while other operations might result in errors.

For more information about management operations in the CS interface and how they affect the LDAP server, see the appendix named "Managing Users, Sites, and Roles in LDAP-Integrated CS Systems" in the *Content Server Administrator's Guide*.

Step 5. Check the `mail` Attribute

Each Content Server user must have a mail attribute (an attribute that stores an e-mail address). Before proceeding, check that the LDAP server's user entries have a `mail` attribute.

Step 6. Create LDAP User Groups (Content Server ACLs)

Use the tools provided by your LDAP server to create groups that correspond to Content Server ACLs. The required groups are listed in this section.

Note

You can create an `ldif` file that contains the user group information from this step, and the user and group membership information from "Step 7. Create Required Users and Their Group Memberships," on page 84. You can then import the `ldif` file into your user directory.

The actual method of importing the `ldif` file varies for each directory, but the structure of the file is standardized among LDAP servers. For examples of `ldif` files, see "Reference: Example ldif Files," on page 90.

Default ALCs (Required for All Installations)

The following list presents the Content Server system default ACLs. You must create groups in the LDAP server whose names exactly match the ACL names below. For information about the access privileges that these ACLs/groups grant, see the "System Defaults" appendix in the *Content Server Administrator's Guid*e.

- Browser
- ContentEditor
- ContentReader
- ElementEditor
- ElementReader
- PageEditor
- PageReader
- RemoteClient
- SiteGod
- TableEditor
- TableReader
- UserEditor
- UserReader
- Visitor
- VisitorAdmin
- xceladmin
- xceleditor

Web Services ACLs

If you are using web services, you must create groups for the following ACLs, making sure that the group names exactly match the ACL names:

- WSAdmin
- WSEditor
- WSUser

For information about the access privileges that these ACLs/groups grant, see the "System Defaults" appendix in the *Content Server Administrator's Guid*e.

Custom ACLs

If any custom ACLs have been created in Content Server since its installation, you must create groups in the LDAP server whose names exactly match the names of the custom ACLs.

Step 7. Create Required Users and Their Group Memberships

Required users include Content Server's default users, sample site users for any sample sites that you will be using, and custom users who are enabled for active sites. Follow instructions in the rest of this step to create the required users and their group memberships.

Default Users

Following is a list of the Content Server and CS-Direct default users. You must re-create the default users in the LDAP server according to the following rules:

- When re-creating default users, you may assign them your choice of names. (Note that the names in the table on page 85 are required by the CS installation process, but not the LDAP server, to populate the Content Server database.)

- Make sure that permissions of the re-created users exactly match the permissions of the users as they are defined in the Content Server database.
- Make the users members of the following groups:

User	Group Memberships
ContentServer (the user that is created in the database during the installation of Content Server)	Browser, ContentEditor, ElementReader, ElementEditor, PageReader, PageEditor, SiteGod, TableEditor, TableReader, UserEditor, UserReader
fwadmin	Browser, ElementEditor, PageEditor, RemoteClient, TableEditor, UserEditor, UserReader, Visitor, VisitorAdmin, xceladmin, xceleditor

To determine which group memberships to grant your additional system users (that is, the actual content providers who will be using this CS system), see the descriptions of the ACLs in the "System Defaults" appendix of the *Content Server Administrator's Guid*e.

Specific details will vary, depending on the structure of your local directory.

Sample Site Users

FatWire recommends that you do not install the sample sites on your CS management or delivery systems. However, if you do install any of the sample sites and intend to use them, be sure that you create the appropriate sample site users in the LDAP server.

Custom Users

If you are modifying an existing installation, you must re-create in the LDAP server all active users, i.e., users who are enabled for Content Server's active sites.

Step 8. (Optional) Create Sites and Roles in the LDAP Server

If you plan to use LDAP attribute mapping to manage your site names and role names in your LDAP server, make sure that you arrange the sites and roles in hierarchical order (for an example see Figure 1, on page 86").

Note

Portal installations use flat-schema LDAP for managing sites and roles. If the portal installations were to also use external LDAP, you would complete the steps in this section, but **would not** create an organizational unit for sites. Instead, you would place the sites and roles at the same level in the LDAP server. Users who are interested in knowing more about LDAP schema can refer to the appendix "Managing Users, Sites, and Roles in LDAP-Integrated CS Systems" in the *Content Server Administrator's Guid*e.

Complete the following steps for hierarchical-schema LDAP:

1. Create an organizational unit for sites.
   a. Under the site unit, create an entry for each of your site names. Note that each name must exactly match the name of the site as you will enter it in the Content Server interface, including spaces, underscores, and so on.

> **Note**
>
> If you are modifying an existing installation to use LDAP, re-create Content Server's active sites and roles in the LDAP server, making sure to name them exactly as in the Content Server interface.

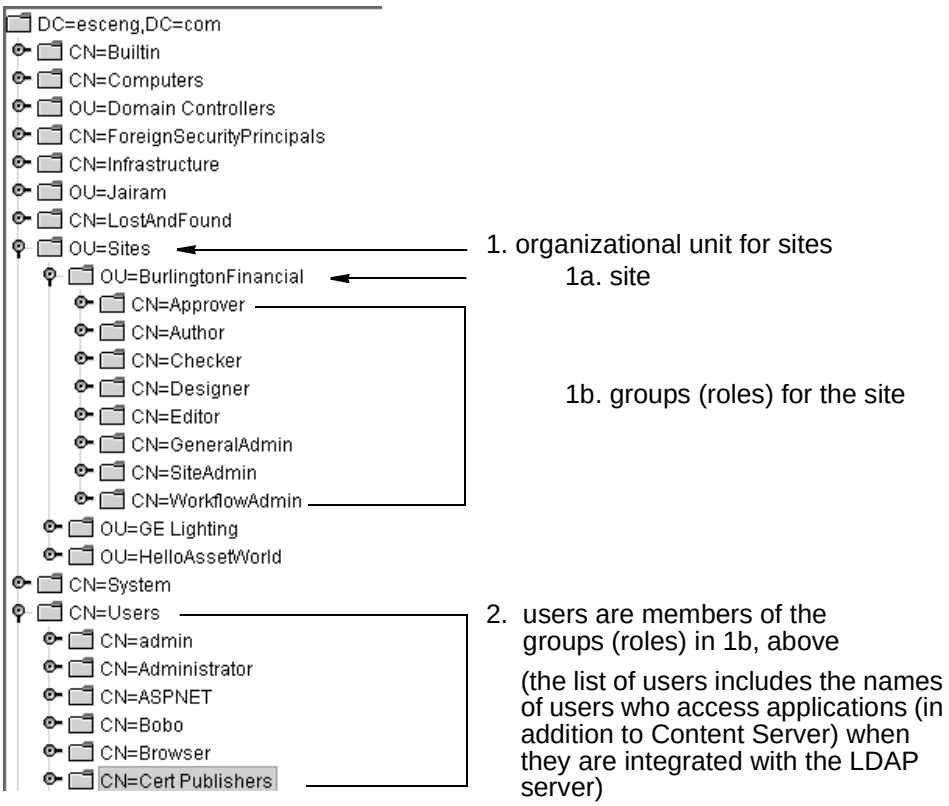
   b. For each site, create a group for each of the appropriate role names. Be sure to add the system default role named GeneralAdmin to each site.

      When you assign any of the system default roles to your sites, be sure that the string exactly matches the names listed here:

      - GeneralAdmin
      - SiteAdmin
      - WorkFlowAdmin

2. For each group, add the appropriate user/members.

Figure 1: LDAP Hierarchies



Step 9. If You Completed Step 8

1. If, in the last step, you created sites and roles in the LDAP server that do not exist in the Content Server database, create the same sites and roles in Content Server, making sure they are named exactly as in the LDAP server. For instructions on creating sites and roles in the CS database, see the following sections and chapter in the *Content Server Administrator's Guid*e:
   - "Creating a Role"
   - Chapter 7, "Assembling and Organizing CM Sites"
2. Enable the users for the sites they are to work in. For instructions, see the section "Granting Users Access to a Site (Assigning Roles to Users)" in the Content Serve*r Administrator's Guid*e.

Step 10. Test the LDAP Integration

To test your integrated system, open a browser and enter the following URL:

`http://`*`servername:port`*`/servlet/CatalogManager?ftcmd=login&`*`username`*`=ContentServer&`*`password`*`=password`

where

`servername` is the name of the machine that is hosting the CS system

`port` is the port number of that server

`username` is user-defined (`Content Server`, in this example)

`password` is user-defined (`password`, in this example)

- If the browser displays a `"Login Successful"` message, you have integrated your LDAP plug-in correctly.
- If you do not see the `"Login Successful"` message, verify that you created the `ContentServer` user in the LDAP server, and that all of the properties were set with the correct values.

Step 11. Resume Your Previous Operations

Once LDAP integration passes the test, you can resume your operations. If they involve the management of ACLs, users, sites, and roles in the LDAP-integrated system, see "Post-Integration Management of ACLs, Users, Sites, and Roles," on page 88 for guidelines.

Post-Integration Management of ACLs, Users, Sites, and Roles

Note

This section applies to all LDAP-integrated systems, regardless of how they were integrated: with bundled LDAP, external LDAP, as web-based systems, or portals.

Once your Content Server system is integrated with LDAP, you are likely to be making changes to custom ACLs and their users, to sites and roles, and to the LDAP configuration. This section offers guidelines for managing these constructs directly in the LDAP server.

ACL and User Management

Access Control Lists, or **ACLs**, are named lists of database operation permissions such as read, write, create, and so on. Because just about everything in Content Server and the CS content applications is represented as a row in a database table, user management on any of your CS systems starts with ACLs.

When using LDAP to manage and authenticate the users of the Content Server system, complete the following steps in the given order:

1. If you are creating ACLs:
   a. Create the ACLs in the LDAP server.
   b. Re-create the ACLs in the Content Server database, using the Content Server interface. Make sure that the names exactly match the names in the LDAP server.
2. Create the **user accounts** with LDAP instead of the Content Server interface:
   a. Create LDAP user groups whose names exactly match the Content Server system default ACLs and any other ACLs that are necessary to implement the design of your system.
   b. Make sure that each user's `mail` attribute exists in the LDAP server.
3. Using the LDAP interface, assign users to the LDAP user groups.

   Content Server assumes that a user who belongs to an LDAP user group also has the Content Server ACL with the corresponding name. For example, a user who belongs to the xceladmin LDAP group has all the access rights granted by the xceladmin ACL.

Site and Role Management

When using LDAP to manage Content Server ACLs and users, you can use the LDAP server to manage sites and roles, as well. In the LDAP interface, you create the sites and roles, and assign the roles to users. Ensure that the same sites and roles exist *in the Content Server database,* and they are named exactly as their LDAP counterparts.

Also, take into account the type of installation you are managing:

- If you are managing the sites and roles of an LDAP-integrated installation that is web-based, you must use hierarchical-schema LDAP. Follow instructions in:
  - "Step 8. (Optional) Create Sites and Roles in the LDAP Server," on page 85 and

  - "Step 9. If You Completed Step 8," on page 87

- If you are managing the sites and/or roles of an LDAP-integrated installation that is portal-based, you must use flat schema LDAP. Follow instructions in:
  - "Step 8. (Optional) Create Sites and Roles in the LDAP Server," on page 85, but **with the following change**: **Do not** create an organizational unit for sites. Place the sites and roles at the same level in the LDAP server.
  - "Step 9. If You Completed Step 8," on page 87

Note

In LDAP-integrated systems, you have the option to manage sites and roles exclusively in the Content Server database, using the Content Server interface. This option bypasses the LDAP server for site and role management operations, as long as the following properties, in `futuretense_xcel.ini`, are set as shown below:

```
xcelerate.usermanagerclass=com.openmarket.xcelerate.user.
  usermanager

xcelerate.rolemanagerclass=com.openmarket.xcelerate.roles.
  rolemanager
```

For more information about site and role management operations in the Content Server interface, see the appendix "Managing Users, Sites, and Roles in LDAP-Integrated CS Systems" in the *Content Server Administrator's Guid*e

If you intend to use the Content Server interface to manage sites and roles, see the following sections and chapters in the *Content Server Administrator's Guid*e:

- "Creating a Role"
- Chapter 7, "Assembling and Organizing CM Sites"
- "Granting Users Access to a Site (Assigning Roles to Users)"

LDAP Configuration Management

If you need to modify CS-LDAP communications (portal or web-based), change the values of the Content Server properties that specify LDAP configuration. Follow instructions in:

- "Step 3. Configure Content Server Properties," on page 79 and
- "Step 4. Configure the LDAP Server," on page 83.

Reference: Example ldif Files

Following are example iPlanet Directory Server `ldif` files that reflect the users and groups for the sample sites. Note that Active Directory `ldif` files will vary from these sample files.

The first file (ldif Example 1 (iPlanet Directory Server): Users) creates both the system default users as well as the example users from the sample sites. The second file (ldif Example 2 (iPlanet Directory Server): Groups) creates groups that match the system default ACLs and assigns them to the appropriate users.

ldif Example 1 (iPlanet Directory Server): Users

```
1  version: 1
2
3  # entry-id: 1
4  dn: ou=people,dc=FatWire,dc=com
5  ou: People
6  objectClass: top
7  objectClass: organizationalunit
8  creatorsName: cn=directory manager
9  modifiersName: cn=directory manager
10 createTimestamp: 20021001153255Z
11 modifyTimestamp: 20021001153255Z
12 nsUniqueId: f2b52281-1dd111b2-80ce8bff-8cf3a98d
13
14 # entry-id: 3
15 dn: uid=DefaultReader,ou=people,dc=FatWire,dc=com
16 givenName: Default
17 sn: Reader
18 objectClass: top
19 objectClass: person
20 objectClass: organizationalPerson
21 objectClass: inetorgperson
22 uid: DefaultReader
23 cn: Default Reader
24 userPassword: {SSHA}UhFRHlAVSIA+RafcjtK/
   Yihz1jq4OrIXexNqaQ==
25 mail: reader@FatWire.com
26 creatorsName: cn=directory manager
27 modifiersName: cn=directory manager
28 createTimestamp: 20021001153259Z
29 modifyTimestamp: 20021001153259Z
30 nsUniqueId: f2b52283-1dd111b2-80ce8bff-8cf3a98d
31
32 # entry-id: 4
33 dn: uid=admin,ou=people,dc=FatWire,dc=com
34 givenName: xcel
35 sn: admin
36 objectClass: top
37 objectClass: person
38 objectClass: organizationalPerson
39 objectClass: inetorgperson
```

```
40 uid: admin
41 cn: admin
42 userPassword: {SSHA}JbN/
   zyGZORP4Wgl+pMvEjOu40bbpVPwMJ2wjUw==
43 mail: admin@FatWire.com
44 creatorsName: cn=directory manager
45 modifiersName: cn=directory manager
46 createTimestamp: 20021001153259Z
47 modifyTimestamp: 20021001153259Z
48 nsUniqueId: f2b52284-1dd111b2-80ce8bff-8cf3a98d
49
50 # entry-id: 5
51 dn: uid=editor,ou=people,dc=FatWire,dc=com
52 givenName: e
53 sn: ditor
54 objectClass: top
55 objectClass: person
56 objectClass: organizationalPerson
57 objectClass: inetorgperson
58 uid: editor
59 cn: e ditor
60 userPassword:
   {SSHA}+nnnA5eMwzOHZXv2wssi7UV1ladaJ+gtzXbkGA==
61 mail: editor@FatWire.com
62 creatorsName: cn=directory manager
63 modifiersName: cn=directory manager
64 createTimestamp: 20021001153259Z
65 modifyTimestamp: 20021001153259Z
66 nsUniqueId: f2b52285-1dd111b2-80ce8bff-8cf3a98d
67
68 # entry-id: 6
69 dn: uid=user_author,ou=people,dc=FatWire,dc=com
70 givenName: user
71 sn: author
72 objectClass: top
73 objectClass: person
74 objectClass: organizationalPerson
75 objectClass: inetorgperson
76 uid: user_author
77 cn: user author
78 userPassword: {SSHA}ZUHls9BJq/
   RZGwP7XqQb3t7OmeoRZboqcjAzZw==
79 mail: author@FatWire.com
80 creatorsName: cn=directory manager
81 modifiersName: cn=directory manager
82 createTimestamp: 20021001153259Z
83 modifyTimestamp: 20021001153259Z
84 nsUniqueId: f2b52286-1dd111b2-80ce8bff-8cf3a98d
85
86 # entry-id: 7
87 dn: uid=user_editor,ou=people,dc=FatWire,dc=com
88 givenName: user
```

```
89 sn: editor
90 objectClass: top
91 objectClass: person
92 objectClass: organizationalPerson
93 objectClass: inetorgperson
94 uid: user_editor
95 cn: user editor
96 userPassword:
   {SSHA}mAGUJkOI0dzHLTzty6KIUe8ZhAzLEygin3ZVCg==
97 mail: user_editor@FatWire.com
98 creatorsName: cn=directory manager
99 modifiersName: cn=directory manager
100 createTimestamp: 20021001153259Z
101 modifyTimestamp: 20021001153259Z
102 nsUniqueId: f2b52287-1dd111b2-80ce8bff-8cf3a98d
103
104 # entry-id: 8
105 dn: uid=user_approver,ou=people,dc=FatWire,dc=com
106 givenName: user
107 sn: approver
108 objectClass: top
109 objectClass: person
110 objectClass: organizationalPerson
111 objectClass: inetorgperson
112 uid: user_approver
113 cn: user approver
114 userPassword:
   {SSHA}y9yvUTef78VQ3mHXoWvyuCsZDr8BXKN3uG18Cw==
115 mail: approver@FatWire.com
116 creatorsName: cn=directory manager
117 modifiersName: cn=directory manager
118 createTimestamp: 20021001153259Z
119 modifyTimestamp: 20021001153259Z
120 nsUniqueId: f2b52288-1dd111b2-80ce8bff-8cf3a98d
121
122 # entry-id: 9
123 dn: uid=user_checker,ou=people,dc=FatWire,dc=com
124 givenName: user
125 sn: checker
126 objectClass: top
127 objectClass: person
128 objectClass: organizationalPerson
129 objectClass: inetorgperson
130 uid: user_checker
131 cn: user checker
132 userPassword: {SSHA}G1KCrgHMdXDfpr/
   fRwJf2FcUU+SdXGNCHnGvdA==
133 mail: checker@FatWire.com
134 creatorsName: cn=directory manager
135 modifiersName: cn=directory manager
136 createTimestamp: 20021001153300Z
137 modifyTimestamp: 20021001153300Z
```

```
nsUniqueId: f2b52289-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 10
dn: uid=user_publisher,ou=people,dc=FatWire,dc=com
givenName: user
sn: publisher
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: user_publisher
cn: user publisher
userPassword: {SSHA}npSN8favjxM7aKYNYF+L2yNB1QWcAxR/
   EiJJcg==
mail: publisher@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b5228a-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 11
dn: uid=user_pricer,ou=people,dc=FatWire,dc=com
givenName: user
sn: pricer
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: user_pricer
cn: user pricer
userPassword: {SSHA}Xz65ezg/
   Kug2tk7897tPS+u35r4zTlAUuwK9UA==
mail: pricer@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b5228b-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 12
dn: uid=user_marketer,ou=people,dc=FatWire,dc=com
givenName: user
sn: marketer
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: user_marketer
cn: user marketer
userPassword:
   {SSHA}vZFpfmvnvCLKbFW118Pljp8fHVVRG8IwXzpsAg==
```

```
mail: marketer@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b5228c-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 13
dn: uid=user_analyst,ou=people,dc=FatWire,dc=com
givenName: user
sn: analyst
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: user_analyst
cn: user analyst
userPassword:
   {SSHA}b508MLDI9plb9xHLPvzTDZFUD0pmLYQaIEahZA==
mail: analyst@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b5228d-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 14
dn: uid=user_expert,ou=people,dc=FatWire,dc=com
givenName: user
sn: expert
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: user_expert
cn: user expert
userPassword: {SSHA}IIEk2beA/
   EezlQNdaCTrTVPMCuSBIoENhGfIOg==
mail: expert@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b5228e-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 15
dn: uid=mirroruser,ou=people,dc=FatWire,dc=com
givenName: mirror
sn: user
objectClass: top
objectClass: person
objectClass: organizationalPerson
```

```
objectClass: inetorgperson
uid: mirroruser
cn: mirror user
userPassword:
   {SSHA}18zvk1VCkQt9N04x0AhsyZvMPCxFcr8nz2iYUQ==
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b5228f-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 16
dn: uid=user_designer,ou=people,dc=FatWire,dc=com
givenName: user
sn: designer
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: user_designer
cn: user designer
userPassword: {SSHA}6H8yq7zbjMB7LTNQTptqWV+xz/
   OtT3BOHkv+KA==
mail: designer@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b52290-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 17
dn: uid=Coco,ou=people,dc=FatWire,dc=com
givenName: C
sn: oco
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: Coco
cn: C oco
userPassword: {SSHA}xVRZ4xtUu0bG0GqtfyhubIUsJ2lVF/
   4fsyUTBg==
mail: Coco@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b52291-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 18
dn: uid=Flo,ou=people,dc=FatWire,dc=com
givenName: F
```

```
sn: lo
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: Flo
cn: F lo
userPassword: {SSHA}E0j8hDF7ZxNkoHSsiNnR5M45BY/
   cXe4IT0j2Iw==
mail: Flo@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b52292-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 19
dn: uid=Moe,ou=people,dc=FatWire,dc=com
givenName: M
sn: oe
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: Moe
cn: M oe
userPassword: {SSHA}Q08M/HspZdMaKTQgrxRQvjtSD4/
   6BtMGHjMQSA==
mail: Moe@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b52293-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 20
dn: uid=Bobo,ou=people,dc=FatWire,dc=com
givenName: B
sn: obo
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: Bobo
cn: B obo
userPassword: {SSHA}9ipG+/
   34B08KZwf4SY2f+z2LBs7nTKhkEgX4KQ==
mail: Bobo@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
```

```
nsUniqueId: f2b52294-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 21
dn: uid=Joe,ou=people,dc=FatWire,dc=com
givenName: J
sn: oe
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
uid: Joe
cn: J oe
userPassword:
   {SSHA}pLJf3weGdcaQFE42XFbSzFMnvO3EfElQTnHyQg==
mail: Joe@FatWire.com
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153300Z
modifyTimestamp: 20021001153300Z
nsUniqueId: f2b52295-1dd111b2-80ce8bff-8cf3a98d

# entry-id: 23
dn: uid=ContentServer,ou=people,dc=FatWire,dc=com
uid: ContentServer
givenName: Content
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
sn: Server
cn: Content Server
userPassword:
   {SSHA}oo1fSpFkq+enbnf0n8DkGoutwdaSaCwdx1C9JQ==
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153511Z
modifyTimestamp: 20021001153511Z
nsUniqueId: 5dfef481-1dd211b2-80ce8bff-8cf3a98d
```

ldif Example 2 (iPlanet Directory Server): Groups

```
version: 1

# entry-id: 2
dn: ou=groups,dc=FatWire,dc=com
ou: Groups
objectClass: top
objectClass: organizationalunit
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153330Z
modifyTimestamp: 20021001153330Z
nsUniqueId: 16786897-1dd211b2-80ce8bff-8cf3a98d
```

```
13
14 # entry-id: 3
15 dn: cn=Browser,ou=groups,dc=FatWire,dc=com
16 objectClass: top
17 objectClass: groupofuniquenames
18 uniqueMember: uid=DefaultReader,ou=people,dc=FatWire,dc=com
19 uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
20 uniqueMember: uid=user_analyst,ou=people,dc=FatWire,dc=com
21 uniqueMember: uid=user_author,ou=people,dc=FatWire,dc=com
22 uniqueMember: uid=user_approver,ou=people,dc=FatWire,dc=com
23 uniqueMember: uid=user_checker,ou=people,dc=FatWire,dc=com
24 uniqueMember: uid=user_editor,ou=people,dc=FatWire,dc=com
25 uniqueMember: uid=user_expert,ou=people,dc=FatWire,dc=com
26 uniqueMember: uid=user_marketer,ou=people,dc=FatWire,dc=com
27 uniqueMember: uid=user_pricer,ou=people,dc=FatWire,dc=com
28 uniqueMember:
   uid=user_publisher,ou=people,dc=FatWire,dc=com
29 uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
30 uniqueMember: uid=editor,ou=people,dc=FatWire,dc=com
31 uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
32 uniqueMember: uid=Bobo,ou=people,dc=FatWire,dc=com
33 uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
34 uniqueMember: uid=Flo,ou=people,dc=FatWire,dc=com
35 uniqueMember: uid=Joe,ou=people,dc=FatWire,dc=com
36 uniqueMember: uid=Moe,ou=people,dc=FatWire,dc=com
37 uniqueMember: uid=ContentServer,ou=people,dc=FatWire,dc=com
38 cn: Browser
39 creatorsName: cn=directory manager
40 modifiersName: cn=directory manager
41 createTimestamp: 20021001153330Z
42 modifyTimestamp: 20021001153712Z
43 nsUniqueId: 16786898-1dd211b2-80ce8bff-8cf3a98d
44
45 # entry-id: 4
46 dn: cn=SiteGod,ou=groups,dc=FatWire,dc=com
47 objectClass: top
48 objectClass: groupofuniquenames
49 cn: SiteGod
50 creatorsName: cn=directory manager
51 modifiersName: cn=directory manager
52 createTimestamp: 20021001153331Z
53 modifyTimestamp: 20021001153734Z
54 nsUniqueId: 16786899-1dd211b2-80ce8bff-8cf3a98d
55 uniqueMember: uid=ContentServer,ou=people,dc=FatWire,dc=com
56
57 # entry-id: 5
58 dn: cn=ContentEditor,ou=groups,dc=FatWire,dc=com
59 objectClass: top
60 objectClass: groupofuniquenames
61 cn: ContentEditor
62 creatorsName: cn=directory manager
63 modifiersName: cn=directory manager
```

```
64 createTimestamp: 20021001153331Z
65 modifyTimestamp: 20021001153851Z
66 nsUniqueId: 1678689a-1dd211b2-80ce8bff-8cf3a98d
67 uniqueMember: uid=ContentServer,ou=people,dc=FatWire,dc=com
68
69 # entry-id: 6
70 dn: cn=ElementReader,ou=groups,dc=FatWire,dc=com
71 objectClass: top
72 objectClass: groupofuniquenames
73 uniqueMember: uid=user_approver,ou=people,dc=FatWire,dc=com
74 uniqueMember: uid=user_pricer,ou=people,dc=FatWire,dc=com
75 uniqueMember: uid=user_editor,ou=people,dc=FatWire,dc=com
76 uniqueMember: uid=user_marketer,ou=people,dc=FatWire,dc=com
77 uniqueMember: uid=user_analyst,ou=people,dc=FatWire,dc=com
78 uniqueMember: uid=user_author,ou=people,dc=FatWire,dc=com
79 uniqueMember: uid=user_expert,ou=people,dc=FatWire,dc=com
80 uniqueMember: uid=user_checker,ou=people,dc=FatWire,dc=com
81 uniqueMember: uid=Bobo,ou=people,dc=FatWire,dc=com
82 uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
83 uniqueMember: uid=Flo,ou=people,dc=FatWire,dc=com
84 uniqueMember: uid=Joe,ou=people,dc=FatWire,dc=com
85 uniqueMember: uid=Moe,ou=people,dc=FatWire,dc=com
86 uniqueMember: uid=ContentServer,ou=people,dc=FatWire,dc=com
87 cn: ElementReader
88 creatorsName: cn=directory manager
89 modifiersName: cn=directory manager
90 createTimestamp: 20021001153331Z
91 modifyTimestamp: 20021001153909Z
92 nsUniqueId: 1678689b-1dd211b2-80ce8bff-8cf3a98d
93
94 # entry-id: 7
95 dn: cn=ElementEditor,ou=groups,dc=FatWire,dc=com
96 objectClass: top
97 objectClass: groupofuniquenames
98 uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
99 uniqueMember: uid=editor,ou=people,dc=FatWire,dc=com
100 uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
101 uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
102 uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
103 uniqueMember: uid=ContentServer,ou=people,dc=FatWire,dc=com
104 cn: ElementEditor
105 creatorsName: cn=directory manager
106 modifiersName: cn=directory manager
107 createTimestamp: 20021001153331Z
108 modifyTimestamp: 20021001153925Z
109 nsUniqueId: 1678689c-1dd211b2-80ce8bff-8cf3a98d
110
111 # entry-id: 8
112 dn: cn=PageReader,ou=groups,dc=FatWire,dc=com
113 objectClass: top
114 objectClass: groupofuniquenames
115 uniqueMember: uid=user_approver,ou=people,dc=FatWire,dc=com
```

```
uniqueMember: uid=user_pricer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_editor,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_marketer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_analyst,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_author,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_expert,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_checker,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Bobo,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Flo,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Joe,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Moe,ou=people,dc=FatWire,dc=com
cn: PageReader
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153331Z
modifyTimestamp: 20021001153331Z
nsUniqueId: 1678689d-1dd211b2-80ce8bff-8cf3a98d

# entry-id: 9
dn: cn=PageEditor,ou=groups,dc=FatWire,dc=com
objectClass: top
objectClass: groupofuniquenames
uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
uniqueMember: uid=editor,ou=people,dc=FatWire,dc=com
uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
cn: PageEditor
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153331Z
modifyTimestamp: 20021001153331Z
nsUniqueId: 1678689e-1dd211b2-80ce8bff-8cf3a98d

# entry-id: 10
dn: cn=UserReader,ou=groups,dc=FatWire,dc=com
objectClass: top
objectClass: groupofuniquenames
uniqueMember: uid=user_approver,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_pricer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_editor,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_marketer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_analyst,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_author,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_expert,ou=people,dc=FatWire,dc=com
uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_checker,ou=people,dc=FatWire,dc=com
uniqueMember: uid=editor,ou=people,dc=FatWire,dc=com
uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Bobo,ou=people,dc=FatWire,dc=com
```

```
168 uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
169 uniqueMember: uid=Flo,ou=people,dc=FatWire,dc=com
170 uniqueMember: uid=Joe,ou=people,dc=FatWire,dc=com
171 uniqueMember: uid=Moe,ou=people,dc=FatWire,dc=com
172 uniqueMember: uid=ContentServer,ou=people,dc=FatWire,dc=com
173 cn: UserReader
174 creatorsName: cn=directory manager
175 modifiersName: cn=directory manager
176 createTimestamp: 20021001153332Z
177 modifyTimestamp: 20021001153956Z
178 nsUniqueId: 1678689f-1dd211b2-80ce8bff-8cf3a98d
179
180 # entry-id: 11
181 dn: cn=UserEditor,ou=groups,dc=FatWire,dc=com
182 objectClass: top
183 objectClass: groupofuniquenames
184 uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
185 uniqueMember: uid=Bobo,ou=people,dc=FatWire,dc=com
186 uniqueMember: uid=ContentServer,ou=people,dc=FatWire,dc=com
187 cn: UserEditor
188 creatorsName: cn=directory manager
189 modifiersName: cn=directory manager
190 createTimestamp: 20021001153332Z
191 modifyTimestamp: 20021001154010Z
192 nsUniqueId: 167868a0-1dd211b2-80ce8bff-8cf3a98d
193
194 # entry-id: 12
195 dn: cn=TableEditor,ou=groups,dc=FatWire,dc=com
196 objectClass: top
197 objectClass: groupofuniquenames
198 uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
199 uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
200 uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
201 uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
202 uniqueMember: uid=ContentServer,ou=people,dc=FatWire,dc=com
203 cn: TableEditor
204 creatorsName: cn=directory manager
205 modifiersName: cn=directory manager
206 createTimestamp: 20021001153332Z
207 modifyTimestamp: 20021001154024Z
208 nsUniqueId: 167868a1-1dd211b2-80ce8bff-8cf3a98d
209
210 # entry-id: 13
211 dn: cn=Visitor,ou=groups,dc=FatWire,dc=com
212 objectClass: top
213 objectClass: groupofuniquenames
214 uniqueMember: uid=user_approver,ou=people,dc=FatWire,dc=com
215 uniqueMember: uid=user_pricer,ou=people,dc=FatWire,dc=com
216 uniqueMember: uid=DefaultReader,ou=people,dc=FatWire,dc=com
217 uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
218 uniqueMember: uid=user_editor,ou=people,dc=FatWire,dc=com
219 uniqueMember: uid=user_marketer,ou=people,dc=FatWire,dc=com
```

```
uniqueMember: uid=user_analyst,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_author,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_expert,ou=people,dc=FatWire,dc=com
uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
uniqueMember: uid=editor,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_checker,ou=people,dc=FatWire,dc=com
uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
cn: Visitor
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153332Z
modifyTimestamp: 20021001153332Z
nsUniqueId: 167868a2-1dd211b2-80ce8bff-8cf3a98d

# entry-id: 14
dn: cn=VisitorAdmin,ou=groups,dc=FatWire,dc=com
objectClass: top
objectClass: groupofuniquenames
uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
cn: VisitorAdmin
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153332Z
modifyTimestamp: 20021001153332Z
nsUniqueId: 167868a3-1dd211b2-80ce8bff-8cf3a98d

# entry-id: 15
dn: cn=RemoteClient,ou=groups,dc=FatWire,dc=com
objectClass: top
objectClass: groupofuniquenames
uniqueMember: uid=user_analyst,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_approver,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_author,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_checker,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_editor,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_expert,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_marketer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_pricer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
cn: RemoteClient
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153332Z
modifyTimestamp: 20021001153332Z
nsUniqueId: 167868a4-1dd211b2-80ce8bff-8cf3a98d

# entry-id: 16
dn: cn=xceladmin,ou=groups,dc=FatWire,dc=com
objectClass: top
```

```
objectClass: groupofuniquenames
uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
uniqueMember: uid=admin,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Bobo,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
cn: xceladmin
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153332Z
modifyTimestamp: 20021001153332Z
nsUniqueId: 167868a5-1dd211b2-80ce8bff-8cf3a98d

# entry-id: 17
dn: cn=xceleditor,ou=groups,dc=FatWire,dc=com
objectClass: top
objectClass: groupofuniquenames
uniqueMember: uid=mirroruser,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_analyst,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_approver,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_author,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_checker,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_designer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_editor,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_expert,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_marketer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=user_pricer,ou=people,dc=FatWire,dc=com
uniqueMember: uid=editor,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Bobo,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Coco,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Flo,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Joe,ou=people,dc=FatWire,dc=com
uniqueMember: uid=Moe,ou=people,dc=FatWire,dc=com
cn: xceleditor
creatorsName: cn=directory manager
modifiersName: cn=directory manager
createTimestamp: 20021001153332Z
modifyTimestamp: 20021001153332Z
nsUniqueId: 167868a6-1dd211b2-80ce8bff-8cf3a98d
```