Oracle8*i*

Administrator's Guide

Release 2 (8.1.6) for Windows NT

January 2000

Part No. A73008-01

ORACLE®

Oracle8*i* Administrator's Guide, Release 2 (8.1.6) for Windows NT

Part No. A73008-01

Primary Authors: Herbert Kelly III, Mark Kennedy, Tamar Rothenberg, and Helen Slattery.

Contributors: Server Technologies Division - NT Development and Robert Farrington of Oracle Support Services.

Contents

1   Oracle8*i* Differences between Windows NT and UNIX

2   Database Tools Overview

3   Multiple Oracle Homes and Optimal Flexible Architecture

4 Using Oracle8*i* Directory Server Features with Active Directory

5   Post-Installation Configuration Tasks

6   Post-Installation Database Creation

7 Administering a Database

8 Authenticating Database Users with Windows

9 Monitoring a Database

10 Tuning Windows NT to Optimize Oracle8*i*

11 Backing Up and Recovering Database Files

12 Developing Applications

A Directory Structures

B Oracle8i Database Specifications for Windows NT

C Oracle8i Configuration Parameters and the Registry

G   Error Messages

Glossary

Index

Contact Us!

Oracle8*i* Administrator's Guide, Release 2 (8.1.6) for Windows NT

Part No. A73008-01

This document describes how to contact Oracle Corporation if you have issues with the documentation or software. It also provides a list of useful resources for Oracle partners and developers.

Read the section...	If you...
"How to Contact Oracle Technical Publications" on page xvi	Have issues with Documentation
"How to Contact Oracle Support Services" on page xvii	Have issues with Software
"Resources for Oracle Partners and Developers" on page xxi	Want to join an Oracle partner or application developer program

How to Contact Oracle Technical Publications

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this guide?
- Do you have suggestions for improvement? Please indicate the chapter, section, and page number (if available).

You can send comments regarding documentation in the following ways:

- Electronic mail - ntdoc@us.oracle.com
- FAX - (650) 506-7370   Attn: Oracle Windows Platforms Server Documentation
- Postal service:
  Oracle Corporation
  Windows Platforms Server Documentation Manager
  500 Oracle Parkway, MS 1OP8,
  Redwood Shores, CA 94065
  USA

If you would like a reply, please provide your name, address, and telephone number.

How to Contact Oracle Support Services

Please copy this form and distribute within your organization as necessary.

Oracle Support Services can be reached at the following telephone numbers and Web sites. The hours of business are detailed in your support contract and the *Oracle Customer Support Guide* in your kit.

Oracle Support Services In...	Call...
United States of America	+ (650) 506-1500 for customers with support contracts.
	+ (650) 506-5577 to obtain a support contract.
Europe	+44 1344 860 160 or the local support center in your country.
All other locations	The telephone number for your country listed at the following Web site:
	`http://www.oracle.com/support/contact_us/sup_hot_phone.html`
	Oracle Support Services telephone numbers are also listed in the *Oracle Customer Support Guide* in your kit.

Please complete the following checklist before you call. If you have this information ready, your call can be processed much quicker.

❏   Your CPU Support Identification Number (CSI Number) if applicable.

❏   The hardware name on which your application is running.

❏ The operating system name and release number on which your application is running.

■ To verify the operating system version on Windows NT, enter the following at the MS-DOS command prompt:

```
C:\> WINMSD
```

The *Windows NT Diagnostics* dialog box displays the operating system and Service Pack version.

❏ The release numbers of the Oracle Server and associated products involved in the current problem. For example, Oracle8*i* Enterprise Edition release 8.1.5.0.0 and Oracle Enterprise Manager release 2.0.0.0.0.

■ To verify the release number of the Oracle Server, connect to the database using a tool such as SQL*Plus. The release number is displayed. For example:

```
Connected to:
Oracle8i Enterprise Edition Release 8.1.6.0.0 – Production
With the Partitioning and Java options
PL/SQL Release 8.1.6.0.0 – Production
```

❏ The third-party software version you are using.

■ To verify an application version, from the application's Help menu, select About...

❏ The exact error codes and messages. Please write these down as they occur. They are critical in helping Oracle Support Services to quickly resolve your problem. Note whether there were no errors reported.

_____

_____

_____

_____

_____

❏ A description of the issue, including:

■ **What happened?** For example, the command used and its result.

_____

_____

_____

■ **When did it happen?** For example, during peak system load, or after a certain command, or after an operating system upgrade. In addition, what was happening when the problem occurred?

_____

_____

_____

■ **Where did it happen?** For example, on a particular system, or within a certain procedure or table.

_____

_____

_____

- **What is the extent of the problem?** For example, production system unavailable, or moderate impact but increasing with time, or minimal impact and stable.

- Did the problem affect one user, several users, or all users?

- Has anything changed? For example, if this is an operation that used to work and now fails, what is different? Can you undo any recent changes, to verify whether they are relevant to the issue?

_____

_____

_____

- **Can the problem be reproduced?** This is a critical question for support analysts. For example, did the problem recur on the same system, under the same circumstances? Can the problem be reproduced on another system? Additionally:

- Does installing a software component fail on all client machines, or just one?

- Do all clients fail to connect to the server, or just one?

- If you are able to restart the server or database, does restarting the database or rebooting the server or client machine (if applicable) make a difference?

_____

_____

_____

❑ Keep copies of the Oracle alert log, any trace files, core dumps, and redo log files recorded at or near the time of the incident. Oracle Support Services may need these to further investigate your problem.

To help analyze problems:

- Archive or delete old alert logs. When the database is started without an alert log, a new one is created. In some cases, if you force the problem to recur with a new alert log, the timestamps for the recorded events may indicate which events are relevant.

- Archive or delete old trace files. To check whether the file was modified, right-click and select Properties. The *Properties* dialog box displays the modification date.

- Check the operating system error logs, especially the System log and Application log. These files are relevant to the Oracle Server. To view these files, from the Start menu, choose Programs > Administrative Tools > Event Viewer, and choose System or Application from the Log main menu.

Resources for Oracle Partners and Developers

This section provides information on partner programs and resources for Oracle database administrators and application developers.

Information Source	Description
Oracle Corporation Home Page `http://www.oracle.com`	This Web site is the starting point for general information on Oracle Corporation.
Alliance Online `http://alliance.oracle.com`	Oracle provides leading-edge technology, education, and technical support that enables you to effectively integrate Oracle into your business. By joining the Oracle Partner Program, you demonstrate to customers that you are committed to delivering innovative Oracle-based solutions and services. The greater your commitment to Oracle, the more we can help you grow your business. It's that simple. The value you derive is associated directly with your level of commitment.

Information Source	Description
Oracle Education `http://education.oracle.com/`	Customers come to Oracle Education with a variety of needs. You may require a complete curriculum based on your job role to enable you to implement new technology. Or you may seek an understanding of technology related to your key area of responsibility to help you meet technical challenges. You may be looking for self-paced training that can be used as an ongoing resource for reference and hands-on practice. Or, you may be interested in an overview of a new product upgrade. Whatever your training need, Oracle Education has the solution.
Oracle Technology Network `http://technet.oracle.com/`	The Oracle Technology Network is your definitive source for Oracle technical information for developing for the Internet platform. You will be part of an online community with access to free software, Oracle Technology Network-sponsored Internet developer conferences, and discussion groups on up-to-date Oracle technology. Membership is free.
Oracle Store `http://oraclestore.oracle.com/`	This is Oracle's online shopping center. Come to this site to find special deals on Oracle software, documentation, publications, computer-based training products, and much more.
Oracle Support Services' Support Web Center `http://www.oracle.com/support/`	Oracle Support Services offers a range of programs so you can select the support services you need and access them in the way you prefer: by telephone, electronically, or face to face. These award-winning programs help you maintain your investment in Oracle technology and expertise. Here are some of the resources available in the Support Web Center:

Oracle*MetaLink*
 	
`http://www.oracle.com/support/`
`elec_sup/index.html`
	Oracle*MetaLink* is Oracle Support Services' premier Web support service. It is available to Oracle*metals* customers (Gold, Silver, Bronze), 24 hours a day, seven days a week.

Oracle*Lifecycle*
 	
`http://www.oracle.com/support/`
`sup_serv/lifecycle/index.html`
	Oracle*Lifecycle* is designed to deliver customized, industry-focused, full life-cycle support solutions that enable industry leaders to use Oracle technology to make smart business decisions, achieve operational excellence, and succeed in their markets.

Information Source	Description

Expert*ONLINE*
 	
`http://www.oracle.com/support/`
`sup_serv/online/index.html`
	Oracle Support Services has launched a new line of services called Expert*ONLINE*. These services provide online database administration for companies looking to supplement their existing DBA staff or fill a DBA role. Services range from Expert*DETECT*, a monitoring, diagnostic, and recommendation service, to Expert*DBA*, a full online database administration service.

Virtual Support Analyst (VSA)
 	
`http://www.oracle.com/support/`
`sup_serv/vsa_start.html`
	VSA is Oracle's Internet e-mail service; it is available to U.S. customers with an Oracle*metals* support agreement. With VSA, you can initiate a request for assistance through e-mail, bypassing the queues you may encounter when using telephone support. VSA also enables you to access Oracle's bug database.

Customer Service
 	
`http://www.oracle.com/support/`
`cus_serv/index.html`
	This site provides resources to make your interactions with Oracle as easy as possible. Among the things you can do are:
	
	■ Learn what is a CPU Support Identification (CSI) number
	
	■ Update your technical contact information
	
	■ Find out whom to contact for invoice and collection issues
	
	■ Request product update shipments
	
	■ Access a glossary of Oracle Support Services terms

U.S. Customer Visit Program
 	
`http://www.oracle.com/support/`
`cus_serv/cus_visit.html`
	This U.S.-based program has been established to help our customers understand and obtain maximum benefit from the support services they have purchased.
	
	The visit typically offers a customized orientation presentation, a comprehensive overview and demonstration of Oracle's electronic services, and helpful tips on working more effectively with Oracle Support Services.

Support Web Center Library
 	
`http://www.oracle.com/support/`
`library/index.html`
	This site contains articles, guides, and other documentation to help you leverage the wealth of knowledge and reference material that Oracle Support Services produces.

Before You Begin

This guide is your primary source of introductory, post-installation, configuration, and administration information for using the products Oracle8*i* Enterprise Edition and Oracle8*i* for Windows NT. This guide incorporates information on both Oracle8*i* Enterprise Edition and Oracle8*i* for Windows NT. Differences between the two versions are noted where appropriate.

Specific topics discussed are:

- Prerequisites
- Intended Audience
- How This Guide Is Organized
- Documentation and Code Conventions

Prerequisites

This guide assumes that you are familiar with the following:

- Windows NT, and have installed and tested it on your computer system

- Object-relational database management concepts

    Additional Information: If you are not familiar with object-relational database management concepts, see *Oracle8i Concepts.*

Intended Audience

This guide is necessary for anyone installing, configuring, or administering Oracle8*i* Enterprise Edition and Oracle8*i* for Windows NT.

Note: This guide describes *only* the features of Oracle8*i* Enterprise Edition and Oracle8*i* software that apply to the Windows NT, Windows 95, Windows 98 and Windows 2000 operating systems. For information about Oracle8*i* Enterprise Edition and Oracle8*i* that is applicable to *all* operating systems, see the other documentation included in your package, listed in Appendix C of *Oracle8i Installation Guide for Windows NT.*

How This Guide Is Organized

This guide is organized as follows:

Chapter 1, "Oracle8i Differences between Windows NT and UNIX"
Provides a list of differences between Oracle8*i* on Windows NT and on UNIX.

Chapter 2, "Database Tools Overview"
Provides a list of preferred and optional tools you can use to perform common database administration tasks.

Chapter 3, "Multiple Oracle Homes and Optimal Flexible Architecture"

Describes how to use multiple Oracle homes and an Optimal Flexible Architecture (OFA) configuration for placement of database files. Read this chapter *before* installing Oracle8*i* Enterprise Edition or Oracle8*i* for Windows NT.

Chapter 4, "Using Oracle8i Directory Server Features with Active Directory"

Describes Oracle integration with the Active Directory.

Chapter 5, "Post-Installation Configuration Tasks"

Describes or references the configuration tasks you may need to perform before using such products as Oracle interMedia.

Chapter 6, "Post-Installation Database Creation"

Describes how to create a database *after* installation with either the Oracle Database Configuration Assistant or the BUILD_DB.SQL script.

Chapter 7, "Administering a Database"

Describes how to administer a database.

Chapter 8, "Authenticating Database Users with Windows"

Describes how to authenticate Oracle8*i* database users with Windows NT.

Chapter 9, "Monitoring a Database"

Describes how to monitor the Oracle8*i* database.

Chapter 10, "Tuning Windows NT to Optimize Oracle8i"

Describes how to tune the Windows NT Server operating system to ensure that your Oracle8*i* database is running in the best possible environment.

Chapter 11, "Backing Up and Recovering Database Files"

Provides recommendations and procedures for backing up and recovering database files.

Chapter 12, "Developing Applications"

Describes Windows NT-specific issues for application developers.

Appendix A, "Directory Structures"

Describes the default directory structures created when you install Oracle components.

Appendix B, "Oracle8i Database Specifications for Windows NT"

Describes Oracle8*i* database initialization parameters, how to edit them, and parameters without Windows NT-specific values. This appendix also explains how to calculate database limits.

Appendix C, "Oracle8i Configuration Parameters and the Registry"

Describes the use of the registry for various Oracle components. In addition, this chapter lists the recommended values and ranges for configuration parameters.

Appendix D, "Storing Tablespaces on Raw Partitions"

Describes how to create raw partitions for database tablespaces.

Appendix E, "Net8 Configuration"

Describes configuration for the Windows NT, Windows 95 or 98 platforms.

Appendix F, "SNMP Support"

Describes Oracle SNMP Agent and SNMP support.

Appendix G, "Error Messages"

Lists the error messages, causes, and corrective actions that are specific to the operation of Oracle8*i* for Windows NT.

Glossary

Provides brief descriptions of terms used throughout this guide.

Documentation and Code Conventions

The following conventions are used in this guide:

Convention	Example	Meaning
All uppercase plain	C:\ORACLE\ORA81	Indicates command names, SQL reserved words, and keywords, as in ALTER DATABASE. All uppercase plain is also used for directory names and file names.
Italic	■ Used to indicate a variable: *file name* ■ Used to indicate the title of a guide.	Indicates a value that you must provide. For example, if a command asks you to type *file name*, you must type the actual name of the file.
Square brackets [ ]	*X*:\[*PATHNAME*]\ORACLE\ *HOME_NAME*	Encloses optional items. For example, when you create an OFA-compliant Oracle home directory, you can place an optional pathname before the \ORACLE pathname. Square brackets also indicate a function key, for example [Enter].
Choose Start >	Choose Start > Programs > Oracle - *HOME_NAME* > Network Administration > Net8 Assistant	How to start a program. For example, to start Net8 Assistant, you must click the Start button on the taskbar and then choose Programs, Oracle - *HOME_NAME* > Network Administration > Net8 Assistant.
C:\>	C:\ORACLE\ORADATA>	Represents the Windows NT command prompt of the current hard disk drive. Your prompt reflects the subdirectory in which you are working. Referred to as the MS-DOS command prompt in this guide.
Backslash (\) before a directory name	\ORADATA	Indicates that the directory is a subdirectory of the root directory.

Convention	Example	Meaning
ORACLE_HOME and *ORACLE_BASE*	Go to the ORACLE_BASE\ORACLE_HOME\RDBMS\ADMIN directory	In releases prior to 8.1, when you installed Oracle8*i* Enterprise Edition or Oracle8*i* for Windows NT, all subdirectories were located under a top level Oracle home directory, that by default was:
		C:\ORANT for Windows NT
		C:\ORAWIN95 for Windows 95/98
		or whatever you may have called your Oracle home.
		In this Optimal Flexible Architecture (OFA)-compliant release, all subdirectories are no longer under a top level *ORACLE_HOME* directory. There is now a new top-level directory called *ORACLE_BASE* that by default is C:\ORACLE. If you install Oracle8*i* Enterprise Edition or Oracle8*i* for Windows NT release 8.1.6 on a clean computer (that is, there is no other Oracle software on the computer), the default settings for the first Oracle home directory is C:\ORACLE\ORA81. If you run Oracle Universal Installer again and install release 8.2.*x*, the second Oracle home directory is called \ORA82. These Oracle home directories are located directly under *ORACLE_BASE*. All directory path examples in this guide follow OFA conventions.
%ORACLE_HOME%	SQL> @%ORACLE_HOME%\ADMIN\DB_NAME\ADHOC\CATALOG.SQL	In SQL*Plus commands, you may see %ORACLE_HOME%. SQL*Plus is able to locate your Oracle Home directory using the %ORACLE_HOME% variable. This convention can be used in Server Manager, SQL*Plus, Export Utility, and Import Utility.
HOME_NAME	Oracle*HOME_NAME*TNSListener	Represents the Oracle home name.
		The home name can be up to sixteen alphanumeric characters. The only special character allowed in the home name is the underscore.

Convention	Example	Meaning
HOME*ID*	HOME0, HOME1, HOME2	Represents a unique registry subkey for each Oracle home directory in which you install products. A new HOME*ID* is created and incremented each time you install products to a different Oracle home directory on one computer. Each HOME*ID* contains its own configuration parameter settings for installed Oracle products.
Symbols	period . comma , hyphen - semicolon ; colon : equal sign = backslash \ single quote ' double quote " parentheses ()	Symbols other than brackets and vertical bars must be entered in commands exactly as shown.

1

Oracle8*i* Differences between Windows NT and UNIX

The following table lists the major differences between Oracle8*i* on Windows NT and on UNIX. For Oracle database administrators moving from a UNIX platform to Windows NT, this information may be helpful in understanding the Windows NT features that are relevant to Oracle.

Feature	On UNIX...	On Windows NT...
Services **See:** Chapter 7, "Administering a Database"	UNIX daemons are similar to services on Windows NT.	Oracle registers a database instance as a service (OracleService*SID*).
		To connect to and use an Oracle instance, an Oracle service is created during the database creation process and associated with the Oracle database. Once a service is created with the Oracle database, the service can run even while no user is logged on. This feature enables server security while running the Oracle database.
		To Access Services:
		By default, services run under the SYSTEM account.
		1. Choose Start > Settings > Control Panel > Services to access the *Services* dialog box. OracleService*SID* and other Oracle services appear here.

Feature	On UNIX...	On Windows NT...
Processes and Threads **See:** Online Help for Oracle Administration Assistant for Windows NT	Each Oracle background process exists as a separate process, for example, ora_dbw0_V816.	All Oracle background, dedicated server, and client processes are threads of the master ORACLE process.
		All the threads of the ORACLE process share resources on Windows NT. This multithreaded architecture is highly efficient, allowing fast context switches with low overhead.
		To View Processes:
		Use the Oracle Administration Assistant for Windows NT to view processes or kill individual threads.
		1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Administration Assistant for Windows NT.
		2. Right-click the *SID*, for example V816, and choose Process Information.
		Note: The Microsoft Management Console (MMC) is launched when the Oracle Administration Assistant for Windows NT is started on Windows NT 4.0. Oracle Corporation has integrated several database administration snap-ins into the MMC.

Feature	On UNIX...	On Windows NT...
File Sizes **See:** Appendix B, "Oracle8i Database Specifications for Windows NT"	UNIX file system (UFS) or journalled file system (JFS). Maximum file size supported by most vendors is now 32 GB. The Oracle block sizes vary between 2-8K.	Oracle can be installed on FAT and NTFS file systems. By default, Oracle runs under the SYSTEM account, which does not have access to NTFS volumes, unless it is granted. The maximum file size for FAT is 4 GB; for NTFS, 16 Exabytes (EB). The Oracle block size is 8K. The maximum number of blocks per data file is 4 million. The maximum number of data files per database depends on block size. When calculating database limits, the total maximum capacity of the database remains the same regardless of the way the bits are split up.
Initialization Parameters: Multiple Database Writers **See:** Appendix B, "Oracle8i Database Specifications for Windows NT"	You can specify more than one database writer process with the initialization parameter DB_WRITERS. Multiple database writers can help, for example, when a UNIX port does not support asynchronous I/O.	DB_WRITERS, which writes dirty buffers to disk, is not supported. Windows NT supplies its own I/O slaves and uses them to see if I/O is complete. Multiple DB_WRITERS might cause synchronization problems.
Direct Writes to Disk **See:** *Oracle8i Concepts* On both platforms, bypassing the file system buffer cache ensures the data is written to disk.	Oracle uses the O_SYNC flag to bypass the file system buffer cache. The flag name depends on the UNIX port.	Oracle bypasses the file system buffer cache completely.
Memory Resources **See:** *Oracle8i Concepts*	The resources provided by the default kernels are often inadequate for a medium or large Oracle database. The maximum size of a shared memory segment (SHMMAX) and maximum number of semaphores available (SEMMNS) may be too low for Oracle recommendations.	Fewer resources are needed for interprocess communication (IPC) because the operating system is thread-based and not process-based. These resources, including shared memory and semaphores, are not adjustable by the user.

Feature	On UNIX...	On Windows NT...
Install Accounts and Groups **See:** Chapter 7, "Administering a Database"	Uses the concept of a DBA group. The root account cannot be used to install Oracle. A separate Oracle account must be created manually.	Oracle must be installed by a Windows NT user name in the Administrator's group. The user name is automatically added to the Windows NT local group ORA_DBA, which receives SYSDBA the privilege. This allows the user to log into the database with the INTERNAL account and not be prompted for a password. **Password Files:** Password files are located in the *ORACLE_BASE\ORACLE_HOME*DATABASE directory and are named PWD*SID*.ORA, where *SID* identifies the Oracle8*i* database instance.
Dynamic Link Libraries (DLLs) **See:** *Oracle8i Concepts*	Shared libraries are similar to the shared DLLs on Windows NT. Object files and archive libraries are linked to generate the Oracle executables. Relinking is necessary after certain operations, such as installation of a patch.	Oracle DLLs form part of the executable at run time, and, therefore, are smaller. DLLs can be shared between multiple executables. Relinking by the user is not supported, but executable images can be modified using the ORASTACK utility. **Modifying Executable Images:** Modifying executable images on Windows NT reduces the chances of running out of virtual memory when using a large SGA or an SGA with thousands of connections. However, Oracle Corporation recommends doing this under the guidance of Oracle Support Services.
Installation **See:** *Oracle8i Installation Guide for Windows NT*	Many manual setup tasks required on UNIX are not required on Windows NT.	You *do not need* to manually: - set environment variables - create a DBA group for database administrators - create a group for users running Oracle Universal Installer - create an account dedicated to installing and upgrading Oracle components

Feature	On UNIX...	On Windows NT...
Multiple Oracle Homes and OFA Using multiple Oracle homes and Optimal Flexible Architecture (OFA) provides many advantages when administering large databases. OFA is implemented on Windows NT and UNIX in the same way. However, differences exist with regard to the following: ■ The top-level names of the OFA directory tree differ between Windows NT and UNIX. However, the main subdirectory and file names are the same on both operating systems. ■ *ORACLE_BASE* directory. ■ No support for symbolic links on Windows NT. **See:** Chapter 3, "Multiple Oracle Homes and Optimal Flexible Architecture"	Multiple Oracle homes on Windows NT is comparable to installation capabilities on UNIX. Environment variables can be set to specify Oracle homes. *ORACLE_BASE* is associated with a UNIX user's environment. **Symbolic Links** Symbolic links are supported. Although everything seems to be in one directory on the same hard drive, files can be on different hard drives if they are symbolically linked or have that directory as a mount point.	*ORACLE_HOME* directories can be located under a single *ORACLE_BASE* directory. *ORACLE_BASE* is defined in the registry (for example, in HKEY_LOCAL_MACHINE \SOFTWARE\ORACLE\HOME0). Do not set *ORACLE_HOME* in the environment (software run from another Oracle home will not work reliably). In fact, beginning in release 8.1.6, the Oracle Universal Installer will reset it. The goal of OFA is to place all Oracle software under one *ORACLE_BASE* directory and to spread the files across different physical drives as your databases increase in size. Oracle Corporation recommends that you use one logical drive to store your database administration files and that you place other files, as needed, on other logical drives in an ORADATA*DB_NAME* directory. For example, for a database named PROD, there are four logical drives: ■ C:\\ contains an Oracle home and the database administration files. ■ F:\\ contains the redo log files. (The F:\\ drive could also represent two physical drives that have been striped to increase performance.) ■ G:\\ contains one of the control files and all of the tablespace files. (The G:\\ drive could also use a RAID Level-5 configuration to increase reliability.) ■ H:\\ contains the second control file. **Symbolic Links** Symbolic links like those on UNIX are not supported, although Microsoft has announced the intention to support them in a near-future release.

Feature	On UNIX...	On Windows NT...
Automatic Startup/Shutdown	**Automatic Startup**	**Automatic Startup**
See: *Oracle8i Administrator's Guide* and Chapter 7, "Administering a Database" in this guide.	Several files and scripts in different directories are used to start an instance automatically.	Set the registry parameter ORA_*SID*_AUTOSTART to TRUE (the default) using an Oracle tool such as ORADIM.
	Automatic Shutdown	1. Enter the following with parameters at the MS-DOS command prompt: `C:\> ORADIM PARAMETERS`
	Scripts are run on computer shutdown, allowing applications such as Oracle to be shut down cleanly.	2. To start the listener automatically, set the service startup type to automatic.
		Automatic Shutdown
		1. Set the registry parameters ORA_SHUTDOWN and ORA_*SID*_SHUTDOWN to stop the relevant OracleService*SID* and shut down.
		2. Set the registry parameter ORA_*SID*_SHUTDOWNTYPE to control the shutdown mode (the default is I, or Immediate).

Feature	On UNIX...	On Windows NT...
Diagnostic and Tuning Utilities **See:** Chapter 2, "Database Tools Overview" and Chapter 9, "Monitoring a Database"	Performance utilities are not included with the operating system. Utilities such as sar and vmstat are used to monitor Oracle background and shadow processes. These utilities are not integrated with Oracle. Task Manager on Windows NT displays currently running processes and their resource usage, similar to the UNIX ps -ef command or OpenVMS SHOW SYSTEM. However, Task Manager is easier to interpret and the columns can be customized.	Performance utilities include Oracle Performance Monitor, Task Manager, Control Panel, Event Viewer, the registry, User Manager, and Microsoft Management Console (only included with Windows 2000). Oracle is integrated with several of these tools. For example: ■ Oracle Performance Monitor displays key Oracle database information. This tool is the same in appearance and operation as the Windows NT Performance Monitor, except it has been preloaded with Oracle8*i* database performance elements. ■ Event Viewer displays system alert messages, including Oracle startup/shutdown messages and the audit trail.

Feature	On UNIX...	On Windows NT...
Raw Partitions **See:** Appendix D, "Storing Tablespaces on Raw Partitions"	Raw partitions are supported.	Data files for tablespaces can be stored on a file system, or on raw partitions. A raw partition is a portion of a physical disk that is accessed at the lowest possible level. Use the Windows NT Disk Administrator application to create an extended partition on a physical drive. An extended partition points to raw space on the disk that can be assigned multiple logical partitions for the database files. An extended partition avoids the four-partition limit on Windows NT by allowing you to define large numbers of logical partitions to accommodate applications using the Oracle8*i* database. Logical partitions can then be given symbolic link names to free up drive letters. **Oracle Parallel Server** Raw partitions are necessary for the shared data files in an Oracle Parallel Server (OPS) environment, available on both UNIX and Windows NT. OPS, in which Oracle instances run on all nodes simultaneously, provides clustering and high availability.

2

Database Tools Overview

Oracle8*i* for Windows NT includes various tools to perform database functions. This chapter describes the preferred tools to perform common database administration tasks.

Specific topics discussed are:

- Oracle8i Enterprise Edition and Oracle8i

- Choosing a Database Tool

- Starting Database Tools

- Using SQL*Loader

- Using Windows NT Tools

- Optional Windows NT Diagnostic and Tuning Utilities

Oracle8*i* Enterprise Edition and Oracle8*i*

The information in this guide applies to both the Oracle8*i* Enterprise Edition and Oracle8*i* database types. Wherever possible, the name Oracle8*i* for Windows NT is used to describe *both* types. Specific database type names are used only when necessary to avoid confusion. Unless otherwise noted, the features and functionality described in this guide are common to both Oracle8*i* Enterprise Edition and Oracle8*i*.

Choosing a Database Tool

Database tools is a collective term for tools, utilities, and assistants that you can use to perform database administration tasks. Some database tools perform similar tasks, though no one database tool performs all database administration tasks. The following sections indicate which database tools can be used on particular operating systems and the preferred tools to use for common database administration tasks.

Note: This chapter describes tasks that use SQL*Plus command line syntax. In this guide, all Server Manager text and examples have been replaced with SQL*Plus equivalents. Although Server Manager continues to ship with 8.1.*x* releases, Oracle Corporation strongly recommends that you migrate to SQL*Plus as soon as possible. See your SQL*Plus documentation for information on using SQL*Plus to perform database administration tasks.

Note that for all previous Oracle8 8.0.*x* releases, the Server Manager executable was SVRMGR30. For 8.1.6, the Server Manager executable is SVRMGRL. The "L" indicates line mode.

Database Tools and Operating System Compatibility

This table lists database tools and the operating system(s) on which each can be used:

Database Tools	Windows NT and Windows 2000[1]	Windows 95 and Windows 98
Application Development		
SQL*Plus (SQLPLUS)[2]	Yes	Yes
Pro*C/C++	Yes	Yes
Pro*Cobol	Yes	Yes
Object Type Translator (OTT)	Yes	Yes
Oracle Web Publishing Assistant	Yes	No
Oracle Services for Microsoft Transaction Server	Yes	Yes
Oracle WebDB[3]	Yes	Yes
Oracle AppWizard for Microsoft Visual C++	Yes	Yes
Database Administration		
Oracle Administration Assistant for Windows NT	Yes	No
Oracle Database Configuration Assistant	Yes	No
Oracle Enterprise Login Assistant (a feature of Oracle Advanced Security)[4]	Yes	Yes
Oracle Performance Monitor for Windows NT	Yes	No

Database Tools	Windows NT and Windows 2000[1]	Windows 95 and Windows 98
Oracle Enterprise Manager, Release 2.1		
Oracle DBA Management Pack (database tools and wizards)	Yes	Yes
Diagnostics Pack	Yes	Yes
Enterprise Manager:	Yes	Yes
■ Configuration Assistant		
■ Console		
■ Migration Assistant		
Diagnostics Pack	Yes	Yes
Extended Applications	Yes	Yes
■ Application Manager		
■ Replication Manager		
Extended Database Administration:	Yes	Yes
■ Database Configuration Assistant		
■ Data Migration Assistant		
■ Distributed Access Manager, Beta		
■ Enterprise Security Manager		
■ Oracle *inter*Media Text Manager		
■ Oracle Spatial Index Advisor, Beta		
Migration Utilities		
Oracle Data Migration Assistant	Yes	No
Oracle Migration Workbench	Yes	Yes
Oracle Utilities from the MS-DOS Command Line		
Migration Utility (MIG)	Yes	No
DBVERIFY (DBVERF)	Yes	Yes
Export Utility (EXP)	Yes	Yes
Import Utility (IMP)	Yes	Yes
OCOPY	Yes	Yes

Database Tools	Windows NT and Windows 2000[1]	Windows 95 and Windows 98
ORADIM[5]	Yes	No
Password Utility (ORAPWD)[6]	Yes	No
Recovery Manager (RMAN)	Yes	Yes
SQL*Loader (SQLLDR)	Yes	Yes
TKPROF (TKPROF)	Yes	Yes
OPERFCFG	Yes	Yes
Network Administration		
Net**8** Assistant	Yes	Yes
Net**8** Configuration Assistant	Yes	Yes
Oracle Wallet Manager (a feature of Oracle Advanced Security)[7]	Yes	Yes
Windows NT Tools		
Task Manager	Yes	Yes
Control Panel	Yes	Yes
Event Viewer	Yes	No
Registry	Yes	Yes
User Manager	Yes	No
Microsoft Management Console[8]	Yes	No

[1] Windows 2000 production versions were not available during the development and testing of Oracle8i release 8.1.6 products. Oracle Corporation has used the Release Candidate versions of Windows 2000 for development and testing. Refer to the READMEDOC.HTM file at the top level of the CD-ROM for the latest information on certification and support of release 8.1.6 products on Windows 2000.

[2] The ORADEBUG utility can be used through SQL*Plus to send debug commands to Oracle processes. See "Using the ORADEBUG Utility" on page 7-26.

[3] Oracle WebDB is available on a separate CD-ROM.

[4] Available only with Oracle8*i* Enterprise Edition, and not Oracle8*i*.

[5] ORADIM only operates on local databases.

[6] ORAPWD does not work on password files for remote databases.

[7] Available only with Oracle8*i* Enterprise Edition, and not Oracle8*i*.

[8] Automatically included with Windows 2000. To use with Windows NT 4.0, you must obtain the Microsoft Management Console from Microsoft Corporation.

Preferred Database Tools

This table lists common database administration tasks and the various database tools you can use to perform them. Oracle Corporation recommends you use the tools listed in the "Preferred Database Tool" column of the table. After choosing a tool to perform a task, go to "Starting Database Tools" on page 2-8 for instructions on how to start the tool.

Database Administration Task	Preferred Database Tool	Other Database Tools
Create a database	Oracle Database Configuration Assistant	■ ORADIM and SQL*Plus together ■ SQL Worksheet and ORADIM
Delete a database	Oracle Database Configuration Assistant	ORADIM and SQL*Plus together
Delete a database service	Oracle Database Configuration Assistant	ORADIM
Start a database	Instance Manager[1]	■ ORADIM and SQL*Plus together ■ Control Panel > Services ■ SQL Worksheet ■ Oracle Administration Assistant for Windows NT
Shut down a database	Instance Manager	■ ORADIM and SQL*Plus together ■ Control Panel > Services ■ SQL Worksheet ■ Oracle Administration Assistant for Windows NT
Change internal database passwords	ORAPWD	ORADIM[2]
Migrate a database ■ From release 7.*x* to release 8.1.6 ■ From MS SQL Server 6.5 and Sybase Adaptive Server 11	■ Oracle Data Migration Assistant[3] ■ Oracle Migration Workbench	Migration Utility (MIG)
Upgrade a database From release 8.0.*x* to release 8.1.6	Oracle Data Migration Assistant	Run provided scripts in SQL*Plus
Export data	Export Wizard	Export Utility (EXP)
Import data	Import Wizard	Import Utility (IMP)
Load data	Load Wizard	SQL*Loader (SQLLDR)
Publish data to the Web	Oracle WebDB	Oracle Web Publishing Assistant
Back up a database	Backup Wizard	■ Recovery Manager (RMAN) ■ OCOPY[4]

Database Administration Task	Preferred Database Tool	Other Database Tools
Recover a database	Recovery Wizard	■ Recovery Manager (RMAN) ■ OCOPY
Authenticate database administrators and users	Security Manager	■ Oracle Enterprise Login Assistant ■ SQL*Plus ■ Windows NT operating system ■ Oracle Administration Assistant for Windows NT (OS authenticated users)
Grant database roles	Security Manager	■ User Manager ■ Oracle Administration Assistant for Windows NT (OS authenticated users)
Create database objects[5]	Schema Manager	■ SQL*Plus
View processes	Oracle Administration Assistant for Windows NT	■ Task Manager

[1] Instance Manager can start up and shut down a database and perform other limited functions. This tool cannot be used to create database services, or create and delete databases. Instance Manager is an Oracle Enterprise Manager tool.

[2] ORADIM can only set a password when none was previously set. Also, ORADIM can change a password by deleting and recreating the Oracle8*i* services. See Chapter 9, "Monitoring a Database" for more information.

[3] Oracle Data Migration Assistant can only be used to upgrade release 7.x or 8.0.x databases to release 8.1.6. It cannot be used to upgrade an earlier Oracle7 database release to a later Oracle7 database release. See your Oracle7 for Windows NT documentation for information on how to upgrade from an earlier Oracle7 release to a later Oracle7 release.

[4] Do not back up files while you are shutting down the database, otherwise your backup will be invalid. You cannot use an invalid backup to restore files at a later date. See *Oracle8i Backup and Recovery Guide* for more information.

[5] See *Oracle8i Administrator's Guide* for guidelines on creating databases objects. This guide provides equations for estimating the space requirements for clusters, non-clustered tables, and indexes. Windows NT uses the same fixed header, transaction header, and row header constants described in this guide.

Starting Database Tools

This section describes how to start each of the database tools in the following categories:

- Starting Database Tools in Multiple Oracle Homes

- Starting Database Tools

- Starting Oracle Utilities from the Command Line

- Starting Oracle Enterprise Manager

- Starting Windows NT Tools

You will be referred back to this section for database tool startup procedures as you use this guide.

Starting Database Tools in Multiple Oracle Homes

If you have multiple Oracle homes on your computer from previous releases, see "Multiple Oracle Home Functionality in Different Releases" on page 3-3 and "Multiple Oracle Home Environments" on page 3-5 for a description of the differences between pre-8.1.6 Oracle homes and release 8.1.6 and later Oracle homes.

Starting Tools from Release 8.0.4 and later 8.0.*x* Multiple Oracle Homes

If you are using multiple Oracle homes functionality, the command to start a tool includes a *HOME_NAME*, where *HOME_NAME* indicates the name of a different Oracle home. Note that *the first* Oracle home created on your computer does not have *HOME_NAME* appended to the group. For example:

To start Oracle Administration Assistant from the first Oracle home, choose:

Start > Programs > Oracle > Database Administration > Oracle Administration Assistant for Windows NT

To start Oracle Database Assistant from an additional Oracle home, choose:

Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Administration Assistant for Windows NT

Starting Tools from Release 8.1.6 Multiple Oracle Homes

In release 8.1.6, all Oracle homes, including the first Oracle home you create on your computer, have a unique *HOME_NAME*. For example, the command to start Database Configuration Assistant is as follows:

Start > Programs > Oracle - *HOME_NAME* > Database Administrations > Database Configuration Assistant, where *HOME_NAME* is the name of the Oracle home. For example, either HOME1 or HOME2 in the following figure:

Taskbar pathname for first Oracle database installation. A home name of HOME1 appears in the pathname.

Taskbar pathname for second Oracle database installation. A home name of HOME2 appears in the pathname.
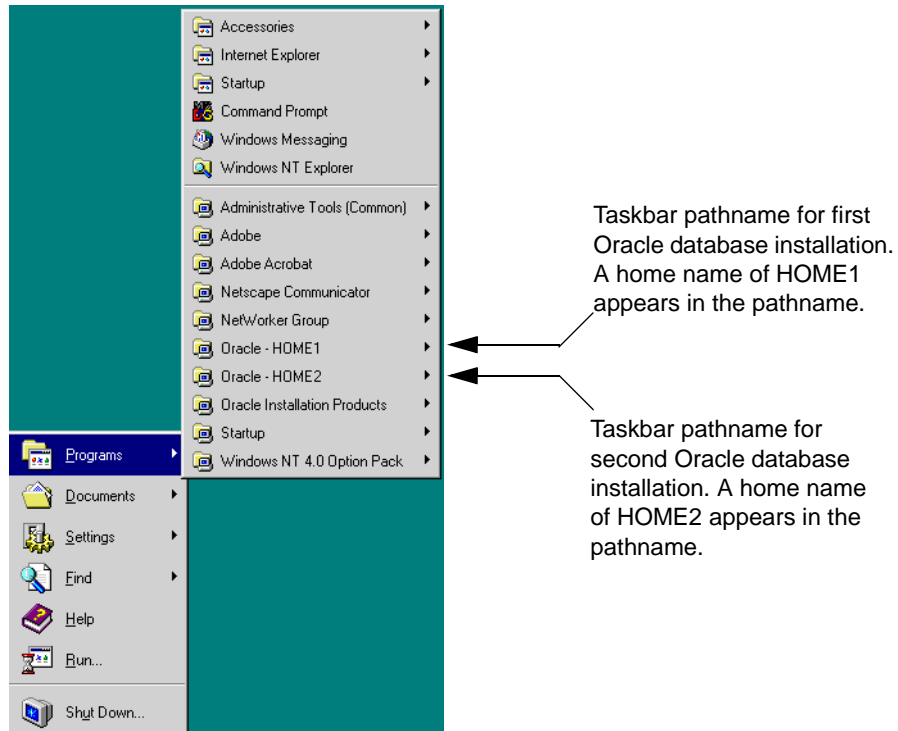
Starting Database Tools

This table describes how to start most tools, and where to go for further information on using these products[1]:

Tool	Choose Start > Programs > Oracle - *HOME_NAME* >	For More Information, See...
Oracle Data Migration Assistant	Extended Database Administration > Data Migration Assistant	■ *Oracle8i Migration* ■ *Oracle8i Installation Guide for Windows NT*
Oracle Migration Workbench	Migration Utilities > Migration Workbench	■ Oracle Migration Workbench Release Notes ■ *Oracle Migration Workbench for MS SQL Server and Sybase Adaptive Server Reference Guide* ■ *Oracle Migration Workbench for MS Access Reference Guide*
Oracle Database Configuration Assistant	Extended Database Administration > Database Configuration Assistant	■ "Using Oracle Database Configuration Assistant" on page 6-4 ■ *Oracle8i Installation Guide for Windows NT*
Net8 Assistant	Network Administration > Net8 Assistant	*Net8 Administrator's Guide*
Net8 Configuration Assistant	Network Administration > Net8 Configuration Assistant	*Net8 Administrator's Guide*
Oracle Wallet Manager	Network Administration > Wallet Manager	*Oracle Advanced Security Administrator's Guide*
Oracle Web Publishing Assistant	Application Development > Oracle Web Publishing Assistant	*Oracle Web Publishing Assistant Getting Started*
Oracle Administration Assistant for Windows NT	Database Administration > Oracle Administration Assistant for Windows NT	"Administering External Users and Roles" on page 8-14

Tool	Choose Start > Programs > Oracle - *HOME_NAME* >	For More Information, See...
Oracle WebDB	Start your Web browser. In the Address or Location field of your browser, type the URL of your WebDB home page.	*Oracle WebDB Installation Guide and Tutorial*
Oracle Enterprise Login Assistant	Network Administration > Enterprise Login Assistant	*Oracle Advanced Security Administrator's Guide*

[1]When you use an assistant, you must have read/write access to the directory where database files will be moved/created. Additionally, users must have administrative privileges to create an Oracle8*i* database. If the Oracle Database Configuration Assistant is run from an account that is not part of the Administrator's group, the tool exits without completing the operation.

Starting Oracle Utilities from the Command Line

This table describes how to start Oracle utilities from the MS-DOS command line, and where to go for further information on using these products:

Oracle Utilities	To Start...	For More Information, See...

DBVERIFY (DBV)	Enter the following at the MS-DOS command prompt:
	
	`C:\> DBV`
	
	DBVERIFY starts and prompts you for a file name parameter. To obtain a list of parameters, enter the following at the MS-DOS command prompt:
	
	`C:\> DBV HELP=Y`
		Oracle8i Utilities

Export Utility (EXP)	Enter the following at the MS-DOS command prompt followed by your user name and password:
	
	`C:\> EXP`
	
	EXP starts and prompts you for parameters. To obtain a list of these parameters, enter the following at the MS-DOS command prompt:
	
	`C:\> EXP HELP=Y`
	
	Note: When running the Export Utility, the default values for the following parameters under Windows NT are:
	
	BUFFER 4 KB
	
	RECORDLENGTH 2 KB
	
	Note: To export an entire database, you must use the user name SYSTEM. Do not use INTERNAL or SYS.
		■ *Oracle8i Utilities*, which describes how to use the Export Utility
		
		■ *Oracle8i Error Messages* for information on error messages

Oracle Utilities	To Start...	For More Information, See...

Import Utility (IMP)	Enter the following at the MS-DOS command prompt followed by your user name and password:
	
	`C:\> IMP`
	
	IMP starts and prompts you for parameters. To obtain a list of these parameters, enter the following at the MS-DOS command prompt:
	
	`C:\> IMP HELP=Y`
	
	Note: When running the Import Utility, the default values for the following parameters under Windows NT are:
	
	BUFFER 4 KB
	
	RECORDLENGTH 2 KB
		■ *Oracle8i Utilities*, which describes how to use the Import Utility
		
		■ *Oracle8i Error Messages* for information on error messages

Migration Utility (MIG)	Enter the following at the MS-DOS command prompt:
	
	`C:\> MIG`
	
	To obtain a list of parameters, enter the following at the MS-DOS command prompt:
	
	`C:\> MIG HELP=Y`
		Oracle8i Migration

OCOPY	Enter the following at the MS-DOS command prompt: `C:\> OCOPY`	"Backing Up Files with OCOPY" on page 11-4
OPERFCFG	Enter the following at the MS-DOS command prompt: `C:\> OPERFCFG`	"Modifying Oracle Performance Monitor for Windows NT Parameters" on page C-20
ORADIM	Enter the following *with* parameters at the MS-DOS command prompt: `C:\> ORADIM PARAMETERS` To get a complete listing of ORADIM parameters, enter the following: `C:\> ORADIM -?	-H	-HELP` **Note:** Entering ORADIM without any options also displays a listing of parameters.	"Using ORADIM to Administer an Oracle Instance" on page 6-26
Password Utility (ORAPWD)	Enter the following at the MS-DOS command prompt: `C:\> ORAPWD` Note that the password file is a hidden file. To see it in a file list, from the Windows NT Explorer, choose View > Options > View > Show All Files	"Creating Password Files" on page 7-16

Oracle Utilities	To Start...	For More Information, See...
Recovery Manager (RMAN)	Enter the following at the MS-DOS command prompt: `C:\> RMAN PARAMETERS`	■ Recovery Manager (RMAN) in command line mode on page 11-2 and *Oracle8i Backup and Recovery Guide* for instructions on using this tool
SQL*Plus (SQLPLUS)	Enter the following at the MS-DOS command prompt: `C:\> SQLPLUS`	■ *SQL*Plus User's Guide and Reference* ■ "Starting and Shutting Down a Database with SQL*Plus" on page 7-11, for examples of starting and stopping the database with SQL*Plus
SQL*Loader (SQLLDR)	Invoke SQL*Loader at the MS-DOS command prompt followed by certain keywords. Enter the following and SQL*Loader displays a Help screen with the available keywords and default values: `C:\> SQLLDR`	■ *Oracle8i Utilities*, which describes how to use SQL*Loader ■ *Oracle8i Error Messages* for information on error messages ■ "Starting Windows NT Tools" on page 2-17
TKPROF (TKPROF)	Enter the following at the MS-DOS command prompt: `C:\> TKPROF`	*Oracle8i Tuning*

Starting Oracle Enterprise Manager

Applications in the DBA Management Pack can be launched through the Oracle Enterprise Manager console or launched separately as stand-alone applications. All database applications can also be launched from the console within a web browser.

When an application is launched through the console, it is connected to the Oracle Management Server and is used in the Oracle Enterprise Manager repository. When an application is launched separately, the user has the option to connect to either a specific database or to a Management Server. When connected to an Oracle Management Server, the DBA Management Pack application has access to all the databases in that Oracle Enterprise Manager repository.

To start an Oracle Enterprise Manager tool as a stand-alone application:

1. Choose Start > Programs > Oracle - *HOME_NAME* > DBA Management Pack > *tool*.

   For example, choose Start > Program > Oracle - HOME1 > DBA Management Pack > Schema Manager.

After launching a DBA application, the *Oracle Enterprise Manager Login* dialog box appears, giving you the option to connect to either the Oracle Management Server or directly to a single database.
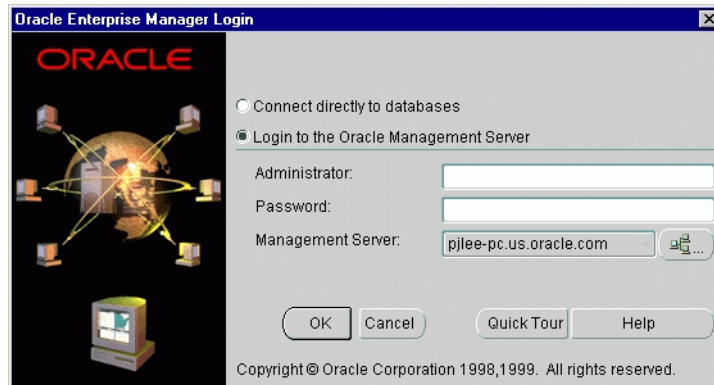
2. Choose the login method for the database administration application and enter the appropriate connect information.

- Login to the Oracle Management Server.

    When a database administration application is connected to the Oracle Management Server, the DBA application can access all the databases on discovered nodes in that repository, and all of these databases appear in the client's tree list of managed objects. The Oracle Management Server must be running for a DBA application to connect to it.

- Connect directly to a single database.

    When connected to a single database, the Oracle Management Server does not need to be running, and that database is the only database that shows in the client's tree list. Your TNSNAMES.ORA file must have an entry for the database. You can also enter the host:port:sid connect string for your service.

    See: See the *Oracle Enterprise Manager Configuration Guide* for information on configuration tasks you must perform before using Oracle Enterprise Manager and information on how to connect to an Oracle database.

To start an Oracle Enterprise Manager tool from the Console:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Enterprise Manager > Console.

   The *Login Information* dialog box appears.

2. Log on when prompted.

3. You can now either:

   ■ Select the database you want to administer in the Navigator tree or in the Map window, then choose the tool from the Console Tools menu or in the Launch Palette.

   ■ Select the database you want to administer in the Navigator tree, then choose the tool from the Related Tools menu of the context-sensitive menu.

   ■ Choose the application from the Console Tools menu or from the Launch Palette, then enter the connect information in the *Login Information* dialog box.

   > **Note:** When you select a database before starting a tool, you are connected to the database according to the preferred credentials that have been set up for the database or the credentials you used to log on to the Console. If connection to the database fails for any reason, the *Login Information* dialog box reappears.

To start an Oracle Enterprise Manager tool from a Web browser:

> **See:** The *Oracle Enterprise Manager Configuration Guide* for information on installing the Oracle Enterprise Manager Web Site, and installing and configuring the Web server.

1. Launch your Web browser and enter the following URL regardless of which Web server you have installed.

   ```
   http://<webserver hostname>:<port number>/
   oem_webstage/EMWebSite.html
   ```

   For example:

   ```
   http://jfox-sun:3339/oem_webstage/EMWebSite.html
   ```

   > **Note:** The Oracle Application Server Listener port number is 3339.

An index page appears, allowing you to launch various products, documentation, and Web sites.

2. Enter the machine name for the Management Server to which you want to connect and click the application icon or name of the application you want to launch.
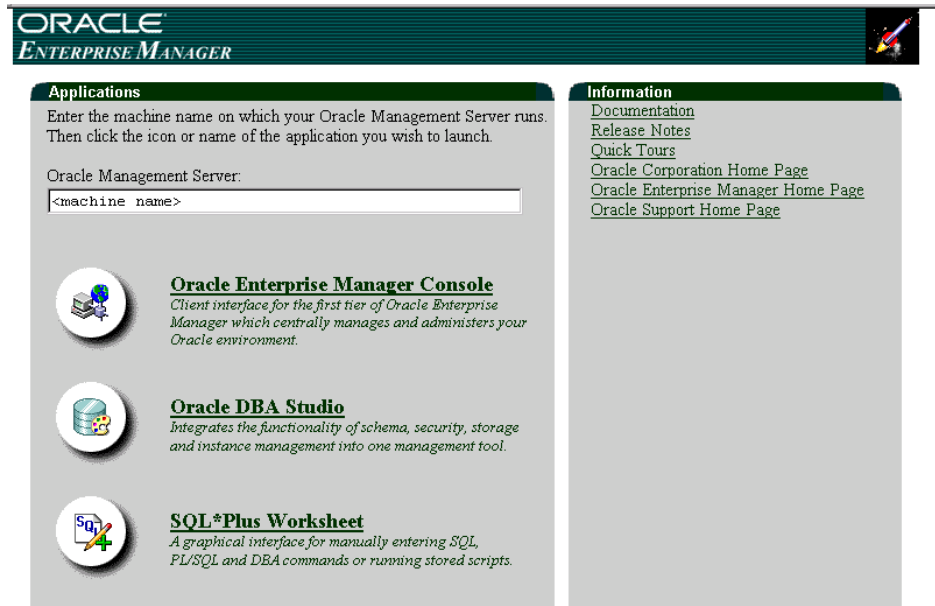
3. If you are logging in to Oracle Enterprise Manager for the first time, type in the default credentials (administrator name and password).

Administrator = `sysman`

Password = `oem_temp`

These credentials are for the default super administrator account. The first time you start Enterprise Manager, you must log in as the super administrator. After other administrator accounts have been created using the super administrator account, you can log in as a different administrator.

Starting Windows NT Tools

This table describes how to start each Windows NT tool, and where to go for more information on using these products:

Windows NT Tools	To Start...	For More Information, See...
Control Panel	Choose Start > Settings > Control Panel	■ "Control Panel" on page 2-20 ■ "Oracle Service Naming Conventions for Multiple Oracle Homes" on page 7-2 ■ Your Microsoft Windows NT documentation
Event Viewer	Choose Start > Programs > Administrative Tools > Event Viewer	■ "Event Viewer" on page 2-21 ■ "Using the Event Viewer" on page 9-9 ■ Your Microsoft Windows NT documentation
Oracle Performance Monitor for Windows NT	Choose Database Administration > Oracle for Windows NT Performance Monitor	■ "Oracle Performance Monitor for Windows NT" on page 2-23 ■ "Using Oracle Performance Monitor for Windows NT" on page 9-2 ■ Your Microsoft Windows NT documentation

Registry	■ Enter the following at the MS-DOS command prompt on Windows NT:
	`C:\> REGEDT32`
	The registry editor window appears.
	■ Enter the following at the MS-DOS command prompt on Windows 95 or Windows 98:
	`C:\> REGEDIT`
	The registry editor window appears.
		■ "Registry" on page 2-25
		■ Appendix C, "Oracle8i Configuration Parameters and the Registry"
		■ Your Microsoft Windows NT documentation

Windows NT Tools	To Start...	For More Information, See...
User Manager	Choose Start > Programs > Administrative Tools > User Manager	■ "User Manager" on page 2-26 ■ Chapter 8, "Authenticating Database Users with Windows" ■ Your Microsoft operating system documentation
Microsoft Management Console (MMC)	Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Administration Assistant for Windows NT **Note**: MMC is launched when the Oracle Administration Assistant for Windows NT is started.	■ Your Microsoft operating system documentation
Task Manager	Right-click on the Task Bar.	■ Your Microsoft operating system documentation

Using SQL*Loader

This section describes Windows NT-specific information for using SQL*Loader (SQLLDR).

Windows NT Processing Options

These are the possible values for the Operating System Dependent (OSD) file processing specifications string option, referred to in the "SQL*Loader Control File Reference" chapter of *Oracle8i Utilities.*

Processing Option	Description
""[1]	Stream record format in which each record is terminated by a newline character. The maximum record size is 48 KB.
"FIX n"	Fixed record format in which each record is exactly *n* bytes long. If the record is terminated by a newline character, the newline character must be the *n*th byte. Note that the *Oracle8i Utilities* guide refers to this control file option as "RECSIZE".

Processing Option	Description
"VAR xxxx"	Load variable length records. Specify the OSD "VAR *recsizehint*" in the control file for this option to take effect. The *xxxx* gives an estimate of the average record size to SQL*Loader so that it can approximate buffer sizes accurately and not waste memory. The default length is eighty characters. The *xxxx* does *not* specify how many leading bytes of length are included in each record. It only acts as a hint to SQL*Loader. Each record must always be preceded by five ASCII bytes containing the length of the remainder of the record. For example, a record must look like the following: `00024This is a 24 byte string` Any whitespace, carriage returns, or linefeeds at the end of the record are ignored unless specifically included in the byte count in the length field.

[1] Two double quote characters with no space in between.

Direct Path Option

SQL*Loader includes a direct path option that bypasses Oracle8*i* for Windows NT redo log and data verification features, thereby decreasing loading time. Use the direct path option with data files known to be error free.

Control File Conventions

When preparing a SQL*Loader control file (.CTL), you must follow certain syntax and notational conventions. When specifying datatypes in the SQL*Loader control file, note that the default sizes of native datatypes are specific to Windows NT. You cannot override these defaults in the control file.

Native Datatypes	Default Field Length
DOUBLE	8
FLOAT	4
INTEGER	4
SMALLINT	2

See: *Oracle8i Utilities* for a complete list of options and instructions on using SQL*Loader.

Using Windows NT Tools

The following Windows NT tools can be used to administer an Oracle database:

- Control Panel
- Event Viewer
- Oracle Performance Monitor for Windows NT
- Registry
- User Manager
- Microsoft Management Console
- Task Manager

Control Panel

The Control Panel enables you to modify system options such as computer services. A service is an executable process registered in the registry and administered by Windows NT. The registry automatically tracks and records security information for each service you create.

Which Oracle Services Appear in the Control Panel?

When you install the Oracle database and other products, Oracle services are created and displayed in the *Services* dialog box:
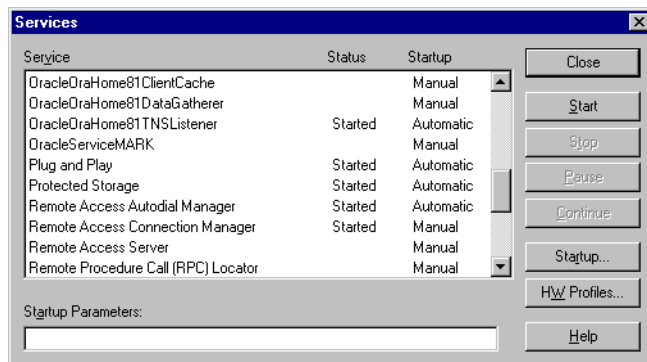
Use the *Services* dialog box to start, stop, pause, or continue each of the Oracle services available on the computer.

Oracle uses services to provide support for its operations, similarly to Windows NT services. In order to create, connect to and use an Oracle instance, an Oracle service is created during the database creation process and associated with the Oracle database.

Once a service is created with your Oracle database, the service can run even while no user is logged on. This is because your Oracle database starts each instance as a service.

> **Additional Information:** You can have multiple, active Oracle home directories on a single computer. This affects the naming conventions for Oracle services. See "Multiple Oracle Home Environments" on page 3-5 and "Oracle Service Naming Conventions for Multiple Oracle Homes" on page 7-2 for additional information.

Event Viewer

Event Viewer is included with the Windows NT operating system, along with the other built-in Windows NT diagnostic and tuning utilities. These include:

- Windows Diagnostics (for obtaining system information)
- Oracle Performance Monitor for Windows NT, described on page 2-23
- Task Manager, described on page 2-28
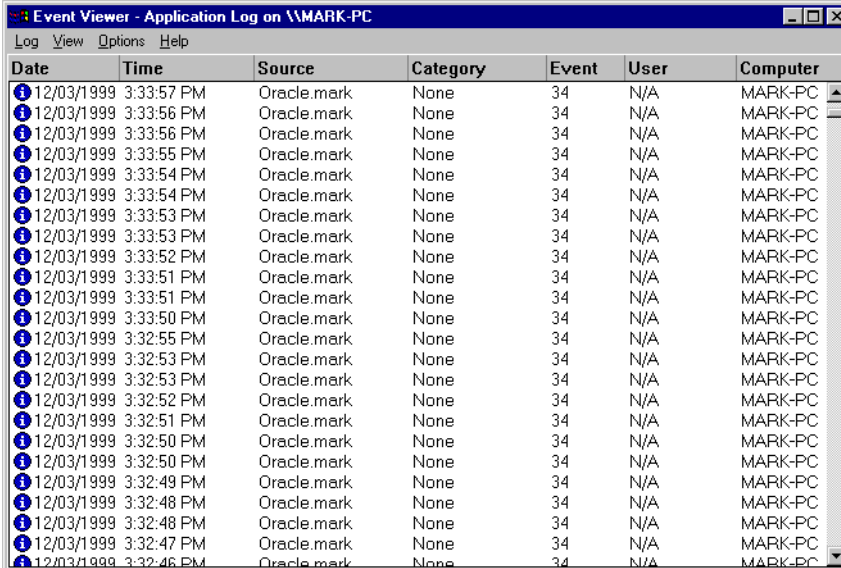
Event Viewer enables you to monitor events in your system. An event is an important occurrence in the system or application (such as your Oracle database) that requires user notification. While messages for major events can display on-screen as you work at your computer, events not requiring your immediate attention are recorded by Windows NT in the Event Viewer log file. You can then view this information at your convenience.

What Oracle Database Events Are Monitored?

Event Viewer can be used to monitor Oracle database events, such as:

- initialization of the System Global Area (SGA) for the active instance
- initialization of the Program Global Area (PGA) for the background processes of the active instance
- connection to the Oracle database with the CONNECT INTERNAL command

In addition, the operating system audit trail is logged to Event Viewer. The following figure shows Event Viewer displaying Oracle database events. Double-click an entry to find out specific information about an event.

> **See:** See Chapter 9, "Monitoring a Database" for specific instructions on accessing and using Event Viewer to monitor Oracle database events.

Oracle Performance Monitor for Windows NT

Oracle Performance Monitor for Windows NT measures computer performance. This tool enables you to view the performance of processors, memory, cache, threads, and processes. Performance information provided includes device usage, queue lengths, delays, throughput measurements, and internal congestion measurements. This information is provided in the form of charts, alerts, and reports.

What Oracle Database Information is Monitored?

Oracle Performance Monitor for Windows NT can be integrated into Microsoft's Performance Monitor. Once this is done, you can use Oracle Performance Monitor to monitor key Oracle database information, such as:

- Library Cache
- Buffer Cache
- Data Dictionary Cache
- Redo Log Buffer Cache
- Thread Activity

You can use your findings to improve database performance. The following figure shows Oracle Performance Monitor monitoring Oracle database objects listed at the bottom of the screen:
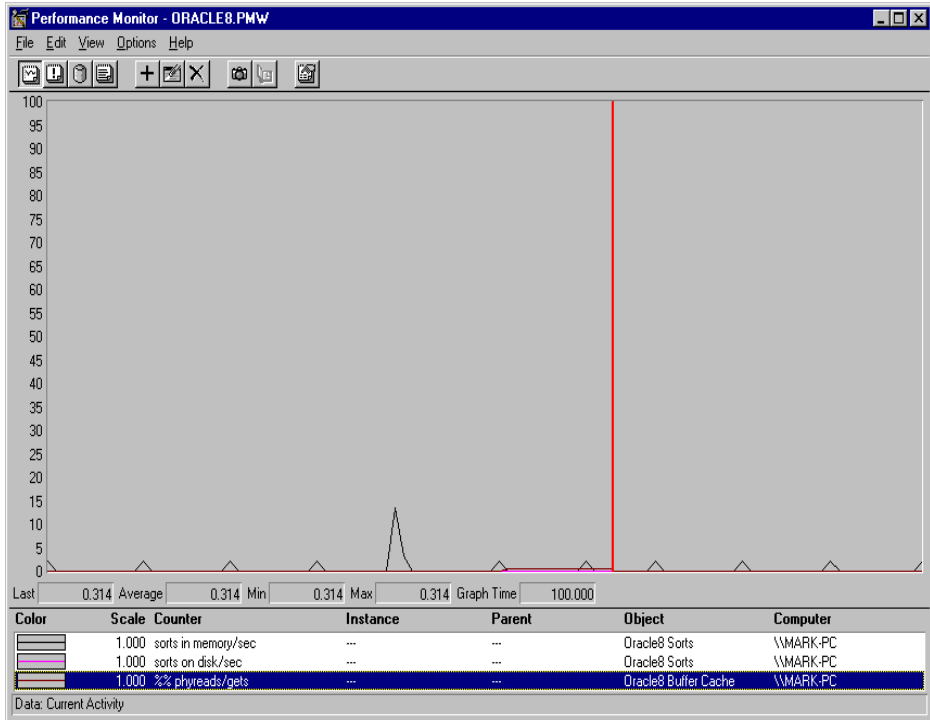
Counters   Objects

> **See:** See "Using Oracle Performance Monitor for Windows NT" on page 9-2 for specific instructions on accessing and using the Oracle Performance Monitor to monitor Oracle database performance.

Registry

The Oracle database stores its configuration information in a structure known as the registry. You can view and modify this configuration information through the registry editor. The registry contains configuration information for your computer, and must not be accessible for editing by inexperienced users. Only experienced administrators should view and change this information.

The registry editor displays configuration information in a tree-like format consisting of four keys (or folders). These keys are shown in the tree view in the left-hand window. In the right-hand window, the parameters and values assigned to that key are displayed.

What Database Parameters Are Configured?

When you install products from your CD-ROM, configuration parameters are automatically entered in the registry. These parameters are read each time your Windows NT computer is restarted and whenever an Oracle product is launched. These parameters include settings for:

- Oracle home directory
- Language
- Company name
- Oracle home subdirectories for individual products
- Individual products such as SQL*Plus
- Services

The following figure shows some of the Oracle database configuration parameters in the registry:
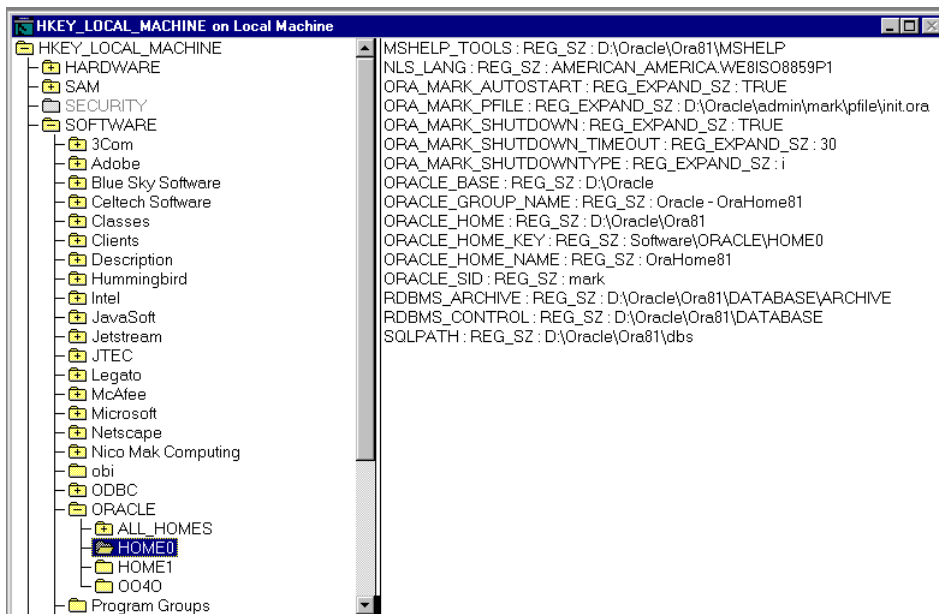
> **See:** See Appendix C, "Oracle8i Configuration Parameters and the Registry" for definitions of Oracle database configuration parameters and specific instructions on using the registry to modify Oracle database configuration parameters.

User Manager

User Manager enables you to manage Windows NT computer security and create user accounts.

What Oracle8*i* Database Tasks Can User Manager Perform?

With User Manager, you can:

- Grant Oracle database roles.

- Use operating system authentication for user accounts. For example, grant DBA access to an NT user.

- Create an NT user account that enables you to make secure client connections to the Oracle8*i* database without a password.

    See: See "Manually Administering External Users and Roles" on on page 8-39 for specific instructions on using User Manager to perform Oracle database administration.

Microsoft Management Console

The Microsoft Management Console provides a central location for network administration programs. The Microsoft Management Console hosts programs (called snap-ins) that administrators can use to manage their networks. Snap-ins run from this central location, helping administrators manage their network products within this single integrated interface.

What Oracle8*i* Database Tasks Can the Microsoft Management Console Perform?

Oracle has integrated several database administration snap-ins into the Microsoft Management Console, which enable database administrators to:

- Configure Oracle database administrators, operators, users, and roles to be authenticated by the Windows NT operating system

- Configure OracleService*SID*

- Modify registry parameters for all Oracle homes on the computer

- Modify the computer hostname, username, and password for the database being monitored by the Oracle Performance Monitor for Windows NT

- View and terminate Oracle threads

The following figure shows the Microsoft Management Console:
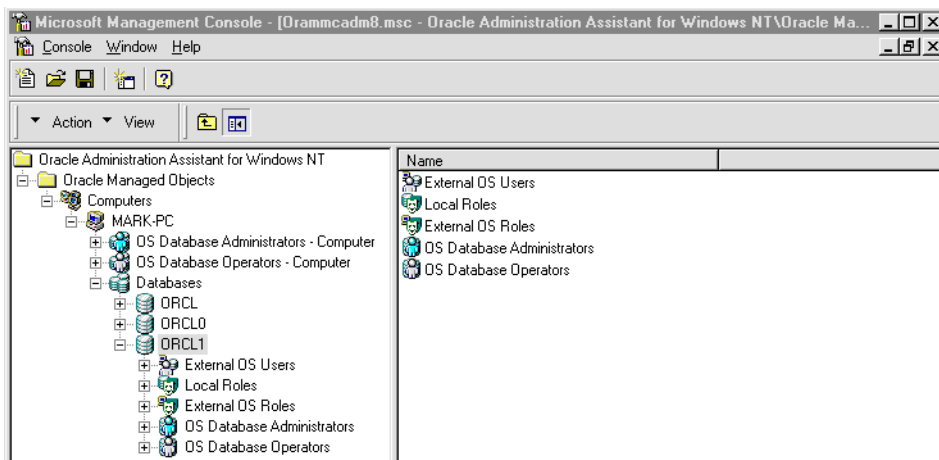
Task Manager

Task Manager has three display options:

- *Applications.* Displays what applications are running. This is useful for identifying and ending unresponsive tasks. (Oracle does not appear as an application because it runs as a service.)

- *Processes.* Displays details of currently-running processes and their resource usage. The columns are customizable.

- *Performance.* Graphically displays sudden CPU and memory usage, which is useful for spotting sudden changes.

Optional Windows NT Diagnostic and Tuning Utilities

The following tools are supplied with the Windows NT Resource Kit:

- **QuickSlice**

  Provides a quick, GUI overview of what is occurring on the system. It has the following benefits:

  - Distinguishes between time spent in user mode and kernel mode

  - Low overhead on the system (unlike Performance Monitor)

  - Shows a continuous display, rather than just a snapshot

  - You can double-click on a process to open a window with more details

- **Process Viewer**

  Summarizes resource usage by a process.

- **Process Explode**

  Provides a detailed display of resource usage by a process.

- **Task List**

  Resource usage and other details of a process may be displayed by giving its PID or process name as an argument to Task List. This tool also displays a list of executables and DLLs associated with a process.

  See Also:

  - Chapter 7, "Administering a Database" for OracleService*SID* configuration tasks

  - Chapter 8, "Authenticating Database Users with Windows" for authentication tasks

  - Chapter 9, "Monitoring a Database" for thread management tasks

  - Appendix C, "Oracle8i Configuration Parameters and the Registry" and Chapter 9, "Monitoring a Database" for Oracle Performance Monitor for Windows NT tasks

  - *Oracle8i Installation Guide for Windows NT*, Chapter 1, "Introducing Oracle8i for Windows NT" for the components available for installation

3

Multiple Oracle Homes and Optimal Flexible Architecture

This chapter describes the concepts of multiple Oracle homes and Optimal Flexible Architecture (OFA) for Oracle8*i* for Windows NT.

Specific topics discussed are:

- Introduction to Multiple Oracle Homes and OFA

- Multiple Oracle Homes Overview

- Which Products Are Multiple Oracle Home-Enabled?

- Changing the Value of PATH

- Exiting Oracle Universal Installer After Entering Name and PATH

- Setting Variables in the Environment or the Registry

- Optimal Flexible Architecture Overview

- Differences Between Directory Trees by Release

- Directory Tree of a Sample OFA-Compliant Database

- OFA Directory Naming Conventions

- OFA and Multiple Oracle Home Configurations

- Increasing Reliability and Performance

- Comparison Between OFA on Windows NT and UNIX

Introduction to Multiple Oracle Homes and OFA

When you install an Oracle database, you are installing one of the largest applications that your computer can support. Using multiple Oracle homes and OFA provides many advantages when administering large databases. The following advantages are the most important:

- Databases are easier to administer because of the structured organization of directories and files, and the consistent naming used for database files.

- A reduction of performance bottlenecks and improved safeguards against disk failures, because input/output (I/O) can be distributed across a number of disks.

- Software upgrades can be tested in an Oracle home in a separate directory from the Oracle home where your production database is located.

Multiple Oracle Homes Overview

This section provides an overview of multiple Oracle homes. It includes the following topics:

- What Is an Oracle Home?

- Benefits of Using Multiple Oracle Homes

- Multiple Oracle Home Functionality in Different Releases

- One-Listener Support of Multiple Oracle Homes

- Multiple Oracle Home Environments

What Is an Oracle Home?

An Oracle home corresponds to the environment in which Oracle products run. This environment includes the following:

- Location of installed product files (for example, C:\ORANT or C:\ORACLE\ORA81)

- PATH variable pointing to the products' binary files

- Registry entries

- Service names

- Program groups

Oracle homes also have a name associated with them, which you specify along with their location during installation.

Benefits of Using Multiple Oracle Homes

The main benefit of using multiple Oracle homes is that you can run multiple releases of the same products concurrently. For example, you can test a release 8.*x.x* database patch before you run your production database release 8.*x.x* against it.

Multiple Oracle Home Functionality in Different Releases

Modifications to multiple Oracle home functionality have occurred since it was introduced in release 8.0.4. This table helps you determine the capabilities of your Oracle home depending on the release you are using.

Release	Oracle Home Functionality
Before 8.0.4	Releases of Oracle for Windows NT and Windows 95 prior to release 8.0.4 only supported single Oracle homes, allowing you to install and run Oracle products in a single Oracle home. Different releases of Oracle products could be installed in the same Oracle home *provided* they had different first or second-digit release numbers. For example, you could install release 7.2 products and release 7.3 products and release 7.*x* and 8.*x* products in the same Oracle home. However, you could not install multiple third-digit releases of the same products. For example, you could not install release 7.3.2 and release 7.3.3 of the same Oracle products on the same computer; one installation would overwrite the other.
8.0.4 to 8.0.6	You can install one or more releases of Oracle products in multiple Oracle homes. For example, with multiple Oracle homes, you can install releases 8.0.*x* and 8.1.3 products or 7.*x* and 8.0.*x* products in different Oracle homes on the same computer. You can also install different releases of Oracle products in the same Oracle home *provided* they have different first or second-digit release numbers. For example, you can install release 7.2 products and release 8.0.*x* products in the same Oracle home.
8.1.3 to 8.1.6	Releases 8.1.3, 8.1.4, 8.1.5, and 8.1.6 have the same multiple Oracle home functionality as release 8.0.4 and later, but with these restrictions: ■ You cannot install releases 8.1.3, 8.1.4, 8.1.5, or 8.1.6 into an Oracle home that was created using the old Installer. (The old Installer was called Oracle Installer and was used for pre-8.1.3 installations; the new Java-based Installer is called Oracle Universal Installer.) ■ You cannot install releases of Oracle prior to release 8.1.3 into an Oracle home that was created by release 8.1.3, 8.1.4, 8.1.5, or 8.1.6. ■ Releases 8.1.3, 8.1.4, 8.1.5, and 8.1.6 must be installed in separate Oracle homes. You cannot have more than one release per Oracle home.
8.1.5 to 8.1.6	You can use a release 8.1.6 listener to spawn a connection to a release 8.1.*x*, 8.0.*x*, or 7.3.*x* database. However, in a mixed environment, you cannot enable the use of shared sockets. Some restrictions exist in using 8.1.6 listeners to spawn connections to earlier versions of the database. These include: ■ You should enable process mode external procedures for release 8.1.6 if you want to spawn a connection to a release 8.0.3 databases. ■ You must install the release 8.0.4.0.3 (or later) patch for Net8. ■ You cannot enable shared sockets.

> **WARNING:** Multiple Oracle homes functionality only works
> with releases 8.0.4 and later. For example, if you have release 7.3.3
> products already installed on your computer, it does not work.
> You cannot install release 7.3.4 products in a separate Oracle
> home.

One-Listener Support of Multiple Oracle Homes

You can use one listener for spawning connections to databases for multiple Oracle homes. You only need to add all the System Identifiers (SIDs) to the SID_LIST section in the *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN\LISTENER.ORA file.

Because the SID is unique to a system across different Oracle homes, the listener can spawn the database thread for a specific SID in the correct Oracle home, and the *ORACLE_HOME* parameter (used in UNIX environments only) is not needed in the LISTENER.ORA.

> **Note:** There may be multiple LISTENER.ORA files on your
> computer, one for each Oracle home. To ensure that you use the
> correct LISTENER.ORA file, check the Oracle home name in the
> listener service. See "Managing Oracle Services" on page 7-2 for
> information on verifying service names.

Multiple Oracle Home Environments

This section describes the differences among multiple Oracle home environments since multiple Oracle homes were first introduced in release 8.0.4.

Release 8.0.4 and Later 8.0.*x* Oracle Home Environments

If you have release 8.0.4 or later 8.0.*x* Oracle homes on your computer, note these differences between the first Oracle home you installed and more recent Oracle homes you may install:

Element	First Oracle Home	Each Additional Oracle Home
Service Names	OracleTNSListener80	Includes the Oracle home name in service names. For example: Oracle*HOME_NAME*TNSListener80
Program Groups	Oracle for Windows NT Oracle home name is not appended to the group.	Appends the Oracle home name to the program group. For example: Oracle for Windows NT - *HOME_NAME*
Registry Entries	Located in HKEY_LOCAL_ MACHINE\SOFTWARE\ORACLE	Subkeys for each Oracle home are added below the HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE subkey (HOME0, HOME1, HOME2, and so on). For more information on the registry keys and subkeys, see Appendix C, "Oracle8i Configuration Parameters and the Registry".
System Identifier (SID) name for starter database	Automatically named ORCL	Only the first starter database on your computer is called ORCL. Additional starter databases use the naming convention ORC*x* or OR*xx* where *x* is a number appended to ensure the SID is unique.

Release 8.1 Oracle Home Environment

Release 8.1 Oracle homes are slightly different from pre-8.1 Oracle homes.

Element	First Oracle Home	Each Additional Oracle Home
Service Names	Oracle*HOME_NAME*TNSListener	Oracle*HOME_NAME*TNSListener
Program Groups	Oracle - *HOME_NAME*	Oracle - *HOME_NAME*
Registry Entries	Located in HKEY_LOCAL_ MACHINE\SOFTWARE\ORACLE\HOME0	Subkeys for each Oracle home are added in the HKEY_LOCAL_ MACHINE\SOFTWARE\ORACLE subkey. For example, the next subkeys after HOME0 are HOME1, HOME2, HOME3, and so on. For more information on the registry keys and subkeys, see Appendix C, "Oracle8i Configuration Parameters and the Registry".
System Identifier (SID)[1] name and DB_NAME	Automatically named ORCL.	For 8.1.3 and 8.1.4, the first starter database on your computer was automatically called ORCL. The second database you created on your computer had a SID of ORCL0. For 8.1.5 and 8.1.6, you must type in the global database name and SID name of your choice when prompted during installation.

[1] For releases 8.1.3 through 8.1.6, the SID can be a maximum of 64 alphanumeric characters in length. For all releases prior to 8.1.3, the SID was a maximum of 4 alphanumeric characters.

Which Products Are Multiple Oracle Home-Enabled?

You can install all products on the CD-ROM into your first Oracle home on a "clean" computer (that is, there is no other Oracle software on the computer) without any conflict.

If you create more Oracle homes, and install the same products that you installed into the first Oracle home, conflicts can arise that cause your original database to function incorrectly if the products are not multiple Oracle home-enabled (multiple Oracle home products.)

To avoid such problems, check the following product classifications before installing multiple versions of the same product on your computer. Oracle products are classified as follows:

- Products Supporting Multiple Oracle Homes

- Products Supporting a Single Oracle Home

- Products Not Supporting Multiple Oracle Homes

- Products Not Associated with an Oracle Home

Products Supporting Multiple Oracle Homes

You can install multiple Oracle home products *multiple times in different Oracle homes.* All products are multiple Oracle home products unless they are listed in:

- "Products Supporting a Single Oracle Home" on page 3-7 or

- "Products Not Supporting Multiple Oracle Homes" on page 3-8

Products Supporting a Single Oracle Home

You can install single Oracle home products into any Oracle home, but *only once per computer.* When installing groups of products, if any of the products in the following list are included in the group and already exist on the computer, do not install them a second time:

- Oracle Performance Monitor for Windows NT

- Oracle Objects for OLE

- Oracle Open Database Connectivity (ODBC) Driver

- Oracle Parallel Server

- Oracle Enterprise Manager

- Oracle SNMP Agent

- All products that depend on any of these products

Products Not Supporting Multiple Oracle Homes

All Oracle7 products and all release 8.0.3 products are non multiple Oracle home products. You can only install these products into an old-style Oracle home (pre-8.0.4 Oracle home is an old-style Oracle home).

Products Not Associated with an Oracle Home

Products not associated with an Oracle home have no restrictions into how many Oracle homes you install them. They include the following:

- Oracle snap-in common files

- Oracle Universal Installer

- Java Runtime Environment

- Oracle Remote Configuration Assistant

When you install these products, Oracle Universal Installer requires that you install them into any Oracle home. However, these files are actually installed in the directory *X:/PROGRAM FILES/ORACLE*, where *X:* is the hard drive where Windows NT is installed.

Changing the Value of PATH

Unless you specify otherwise at installation time, the Oracle home in which you installed products most recently is the first directory listed in your PATH (primary home). As such, it has priority over the other Oracle home entries in your PATH.

If you invoke a product from the MS-DOS command prompt, the release of the product invoked is the one in the Oracle home listed first in your path, unless you specifically invoke a different release of the product by one of the following methods:

- Specifying the full directory path name to the release of the product you want to use at the MS-DOS command prompt.

- Changing to the directory that contains the executable you want to use.

- Changing your PATH so that the first entry points to the binary files for the product release you want to use.

You can change the value of PATH by using one of the following methods:

- Using Oracle Home Selector

- At the System Level

  You can assign a new value at the system level. The new value exists until you change the value of PATH again.

- At the MS-DOS Command Prompt

  You can assign a new value at the MS-DOS command prompt. The new value reverts to its previous value when you quit the session.

> **Note:** The first two methods of changing the value of PATH are only valid if you are a member of the Administrators group. After you have changed the value of PATH, you must open a new MS-DOS window to make it active. The change is not reflected in already-opened MS-DOS windows.

Using Oracle Home Selector

Oracle Home Selector is a graphical user interface (GUI) tool that enables you to edit your environment path to make an appropriate Oracle home directory your primary home. This tool can only be used when you have multiple, active Oracle home directories on a single computer.

To change the value of PATH using Oracle Home Selector:

1. Choose Start > Programs > Oracle Installation Products > Home Selector.

   The *Oracle Home Selector* window appears.

2. Select the Oracle home that you want as the primary Oracle home from the drop-down list.

3. Click OK.

At the System Level

To change the value of PATH at the system level:

On Windows NT

1. Choose Start > Settings > Control Panel.

   The *Control Panel* window appears.

2. Double-click the System icon.

   The *System Properties* window appears.

3. Click the *Environment* tab.

   The system variables appear.

4. Edit the value of PATH in the Value field and click Set.

5. Click OK.

On Windows 95 and Windows 98

1. Open the AUTOEXEC.BAT file.

2. Edit the value of the PATH statement.

3. Reboot your computer.

At the MS-DOS Command Prompt

To change the value of PATH at the MS-DOS command prompt:

At the MS-DOS command prompt, enter:

```
C:\> SET PATH=PATHNAME;%PATH%
```

where *PATHNAME* is the full path to the binary files for the products you want to use. This change is valid for the current session only. If you want to change the value of your PATH more permanently, use Oracle Home Selector or change the value of PATH at the system level. Both methods are described above.

Exiting Oracle Universal Installer After Entering Name and PATH

If you have to exit Oracle Universal Installer unexpectedly after you have entered the name and path for an Oracle home (for example, because there is no more disk space in the path you specified), you cannot specify a different path until you delete the HOME*ID* key and the ID*x* key corresponding to that Oracle home from the registry. To do this:

1. Read the value of the ORACLE_HOME_NAME parameter for each HOME*ID* subkey in the HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE key until you find the value that matches the name of the Oracle home you need to delete.

2. Delete the HOME*ID* subkey you just located.

3. Delete the appropriate ID*x* subkey in the HKEY_LOCAL_ MACHINE\SOFTWARE\ORACLE\ ALL_HOMES key, where *x* has the same value as the *ID* in HOME*ID*. For example, if the HOME*ID* subkey for the home name you want to delete is HOME1, then the appropriate *IDx* subkey is ID1.

> **See Also:** Appendix C, "Oracle8i Configuration Parameters and the Registry" for more information on the registry keys and subkeys.

Setting Variables in the Environment or the Registry

Variables set in the environment always override the value of equivalent variables set in the registry. The following section describes the consequences of setting two of the most commonly-used environment variables, ORACLE_HOME and TNS_ADMIN.

ORACLE_HOME

Oracle Corporation recommends that you *never set* the ORACLE_HOME environment variable because it is not required for Oracle products to function properly. If you set the ORACLE_HOME environment variable, Oracle Universal Installer will unset it for you. Oracle products find the value of ORACLE_HOME at the location specified by the *ORACLE_BASE\ORACLE_HOME*\BIN\ORACLE.KEY file. If there is a need to set ORACLE_HOME in the environment for another reason, care must be taken to only run software from that Oracle home when the variable is set.

When you run an Oracle program from the MS-DOS command prompt, the first executable by that name found in the directory path runs. For example, C:\> SQLPLUS. Alternately, if you specify a full directory path, the specified program runs. For example, C:\ORACLE\ORA81> SQLPLUS.

If you modify the value of PATH using any of the three methods described in the previous section, "Using Oracle Home Selector", "At the System Level", or "At the MS-DOS Command Prompt", you can change the choice of which version of a program is run from the MS-DOS command prompt. In sum, modifying the value of PATH indicates from which Oracle home to run executables, at the MS-DOS command prompt, when no full directory path is specified.

Once an Oracle program starts, it looks for all environment variables in the following order:

1. In the current environment

2. In the registry key for the Oracle home from which the program is running.

   The program knows where it's running from by calling Window NT to obtain the executable's path name, and then parsing the path name to get the directory from which it's running. In the *ORACLE_BASE\ORACLE_HOME*\BIN directory where the executable resides, there is a file called ORACLE.KEY. This file specifies where in the registry to look for variables when programs from that particular Oracle home are run.

For example, if you run C:\ORACLE\ORA81\BIN\SQLPLUS.EXE, SQLPLUS.EXE looks in C:\ORACLE\ORA81\BIN\ORACLE.KEY to find out where to look for its registry variables. If the ORACLE.KEY file does not exist (for version 7.*x* and some version 8.0 Oracle homes), Oracle uses HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE to locate the registry variables.

In a typical case, there are no Oracle variables (that is, ORACLE_HOME) set in the environment. Any programs run from a release 8.0.5 Oracle home look in the ORACLE.KEY file in that Oracle home and find their variables (including ORACLE_HOME) in the correct registry key. Likewise for release 8.1.6, the Oracle home that gets priority depends on the PATH, but regardless of the PATH setting, all the software works correctly.

Consequences of Setting ORACLE_HOME

If you set ORACLE_HOME in the environment, then software run from another Oracle home will not work reliably. The conflict occurs when you set ORACLE_HOME to point to one Oracle home directory, then attempt to run programs from a second Oracle home. These programs first check for any environment variable settings (such as ORACLE_HOME), before checking the registry through the ORACLE.KEY file. Since ORACLE_HOME is set, the programs in the second Oracle home attempt to use files in the first Oracle home, causing a conflict.

For example, assume you have release 8.0.5 installed in C:\ORANT, and release 8.1.6 installed in C:\ORACLE\ORA81, and ORACLE_HOME is set to C:\ORANT in the environment. If you run a program from C:\ORACLE\ORA81\BIN, that program first looks in the environment for all variables *before* looking at its ORACLE.KEY file. So, a program run from your release 8.1.6 Oracle home runs with ORACLE_HOME=C:\ORANT. Therefore, anything that the program uses ORACLE_HOME for will be looked for in C:\ORANT, where it may not exist. Examples include message files (*.MSB), SQL scripts (.SQL), and any other files opened by the program and based off ORACLE_HOME.

Note that the same behavior occurs on UNIX. If you run a program from Oracle home number 1 with ORACLE_HOME=OracleHome number 2 in the environment, then the same behavior can be observed.

TNS_ADMIN

Oracle software looks for TNS_ADMIN in one location in the registry (depending upon the type of Oracle home installed). If you installed software into the default Oracle home, then any software running from that Oracle home will look in HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE. If you installed a new-style (8.0.4 or later) multiple Oracle home, then the Oracle software looks in HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID*. The ALL_HOMES key is used by the installer and plays no role when translating variables.

The environment always overrides the registry, so if TNS_ADMIN is set in the environment, that takes precedence over the TNS_ADMIN setting in the registry. No variables should be set in the environment by the Oracle Home Selector except for the PATH.

Optimal Flexible Architecture Overview

The Oracle Optimal Flexible Architecture (OFA) is a set of file naming and placement guidelines for Oracle software and databases. It can also be thought of as a set of *good habits* to adopt when organizing Oracle directories and files on your computer. All Oracle products on the CD-ROM are OFA-compliant; that is, Oracle Universal Installer places Oracle products in directory locations that follow the OFA guidelines. Although using OFA is not a requirement, Oracle Corporation recommends that you use it if your database will grow in size, or if you plan to have multiple databases.

The aim of OFA is to prevent an entire class of problems that can occur when you have different versions of Oracle software and multiple, growing databases on your computer. OFA is designed to provide significant benefits in the following areas:

- Ease of maintenance of Oracle software and databases through standard file organization
- Reliability through data spanning multiple physical drives
- Performance through decreased I/O contention for disks

For example, one of the many benefits of OFA is that Oracle Universal Installer separates Oracle software executables from database files. Previously, database files were placed in *ORACLE_HOME*\DATABASE, a subdirectory of the Oracle home directory that also contained Oracle software. Using OFA, Oracle Universal Installer puts Oracle software in *ORACLE_BASE**ORACLE_HOME* and database files in *ORACLE_BASE*\ORADATA.

Putting database files in a subdirectory of the Oracle home directory that also contained Oracle software made upgrades unnecessarily difficult. Separating software from data is essential, because over time, when you upgrade a database to the latest release, the new Oracle software executables will be placed in a different Oracle home directory. After the upgrade is judged to have been successful, you can easily remove the old Oracle home directory and reclaim space because the database does not reside there.

Benefits of an OFA-Compliant Database

An OFA-compliant database has the following benefits:

- **Easier database administration and management of database growth**

  The file system is organized to simplify the following tasks:

  - Locating specific database files

  - Adding database files as the database grows

- **Fewer performance bottlenecks**

  Disk contention decreases, because Oracle administration files, binary files, and data files that used to be on one disk can now reside in separate directories or in separate directories on separate disks.

- **Safeguards against disk failures**

  By spreading files across more than one disk, disk failures impact as little data as possible.

- **Support for concurrent execution of application software**

  You can run multiple versions of application software simultaneously, enabling you to test and use a new release of an application before abandoning the previous version. Transferring to a new version after an upgrade is simple for the database administrator and transparent for the user.

Characteristics of an OFA-Compliant Database

An OFA-compliant database has the following characteristics:

- **Independent subdirectories**

  Categories of files are separated into independent subdirectories so that files in one category are minimally affected by operations on files in other categories.

- **Consistent naming conventions for database files**

  Database files are named to realize the following advantages:

  - Database files are easily distinguishable from all other files
  - Files of one database are easily distinguishable from files of another database
  - Control files, redo log files, and data files are easily identifiable
  - Clearly indicated association of data files to tablespaces

- **Integrity of Oracle home directories**

  You can add, move, or delete Oracle home directories without having to revise programs that refer to them.

- **Distinguishes administrative information for each database**

  The ability to separate administrative information about one database from that of another ensures a reasonable structure for the organization and storage of administrative data.

- **Separation of tablespace contents**

  Tablespace contents are separated to realize the following advantages:

  - Minimize tablespace-free space fragmentation
  - Minimize I/O request contention
  - Maximize administrative flexibility

- **Tuning I/O loads across all disks**

  I/O loads are tuned across all disks, including disks storing Oracle data in raw devices, if needed.

Differences Between Directory Trees by Release

OFA has necessitated changes to the Oracle database directory tree. This table lists the differences:

Element	Pre-8.1.3 non-OFA-Compliant	Post-8.1.3 OFA-Compliant
Name of the top-level directory where Oracle is installed.	When you install a pre-8.1.3 release, all subdirectories are located under a top-level *ORACLE_HOME* directory that by default is C:\ORANT. See the following figure on page 3-18 for a depiction of the pre-8.1.3 non-OFA-compliant directory tree.	When you install a post-8.1.3 release, all subdirectories are no longer under a top-level *ORACLE_HOME* directory. There is now a new top-level directory called *ORACLE_BASE* that is of the form *X*:\ORACLE where *X* is any hard drive. If you install an OFA-compliant database using Oracle Universal Installer defaults, *ORACLE_BASE* is C:\ORACLE. *ORACLE_HOME* directories are located under *ORACLE_BASE*. The \ORADATA and \ADMIN directories, which contain the database files and database administration files, are also located under *ORACLE_BASE*. See the following figure on page 3-18 for a depiction of the 8.1.4, 8.1.5, and 8.1.6 OFA-compliant directory tree.
Database file names	Database files have the SID in the database file name. For example, the first control file is named CTL1*SID*.ORA.	Database files no longer have the SID in the database file name. For example, the first control file is named CONTROL01.CTL. There is no need for the presence of the *SID* in the file name because all the database files for a particular database are placed in \ORADATA under a directory called *DB_NAME* that is named for that database.
Database file name extensions	All database files have the same .ORA extension.	The convention of having .ORA as the file name extension for database files is no longer used. Database file names now have more meaningful extensions. These are .CTL for control files, .LOG for log files, and .DBF for data files.

The following figure provides a top-level overview of the old and new database directory trees:

Pre-8.1.3 non-OFA-Compliant Oracle on Windows NT Directory Tree
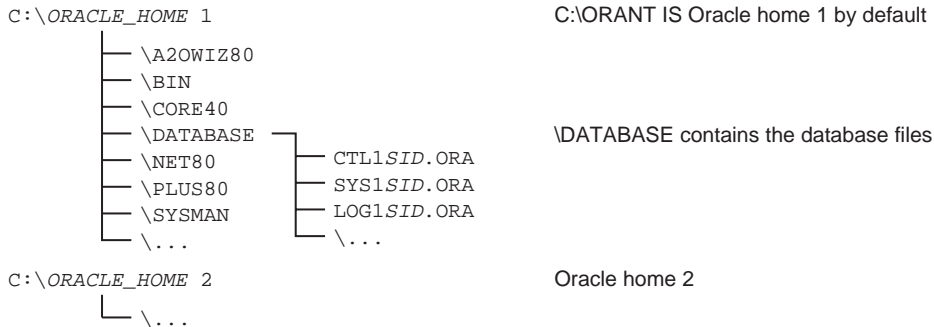
```
C:\ORACLE_HOME 1                                    C:\ORANT IS Oracle home 1 by default

        ─── \A2OWIZ80
        ─── \BIN
        ─── \CORE40
        ─── \DATABASE ───                           \DATABASE contains the database files
        ─── \NET80          ─── CTL1SID.ORA
        ─── \PLUS80         ─── SYS1SID.ORA
        ─── \SYSMAN         ─── LOG1SID.ORA
        ─── \...            ─── \...

C:\ORACLE_HOME 2                                    Oracle home 2
        └── \...
```

8.1.4, 8.1.5, and 8.1.6 OFA-Compliant Oracle on Windows NT Directory Tree

```
C:\ORACLE_BASE                                          C:\ORACLE is ORACLE_BASE by default
        ─── \ORACLE_HOME 1                              \ORA81 is Oracle home 1 by default
                  ─── \BIN
                  ─── \NETWORK
                  └── \ASSISTANTS

        ─── \ORADATA          ─── \ARCHIVE             \ORADATA contains the database files
                  ─── \ORACLE ───
                  ─── \SAMPLE ─── ─── CONTROL01.CTL
                  └── \...         ─── CONTROL02.CTL
                                   ─── SYSTEM01.DBF
                                   ─── REDO01., ...02, ...03, ...04.LOG
                                   ─── INDX01.DBF
                                   └── .CTL, .DBF, .LOG

        ─── \ADMIN                                      \ADMIN contains the administration files
                  ─── \DB_NAME1 ───                      including the INIT.ORA file
                  ─── \DB_NAME2 ─── ─── \ADHOC
                  └── \...          ─── \BDUMP
                                    ─── \PFILE
                                    ─── \...
                                    └── \...

        └── C:\ORACLE_HOME 2                            \ORA82 is Oracle home for release 8.2
                  └── \...
```
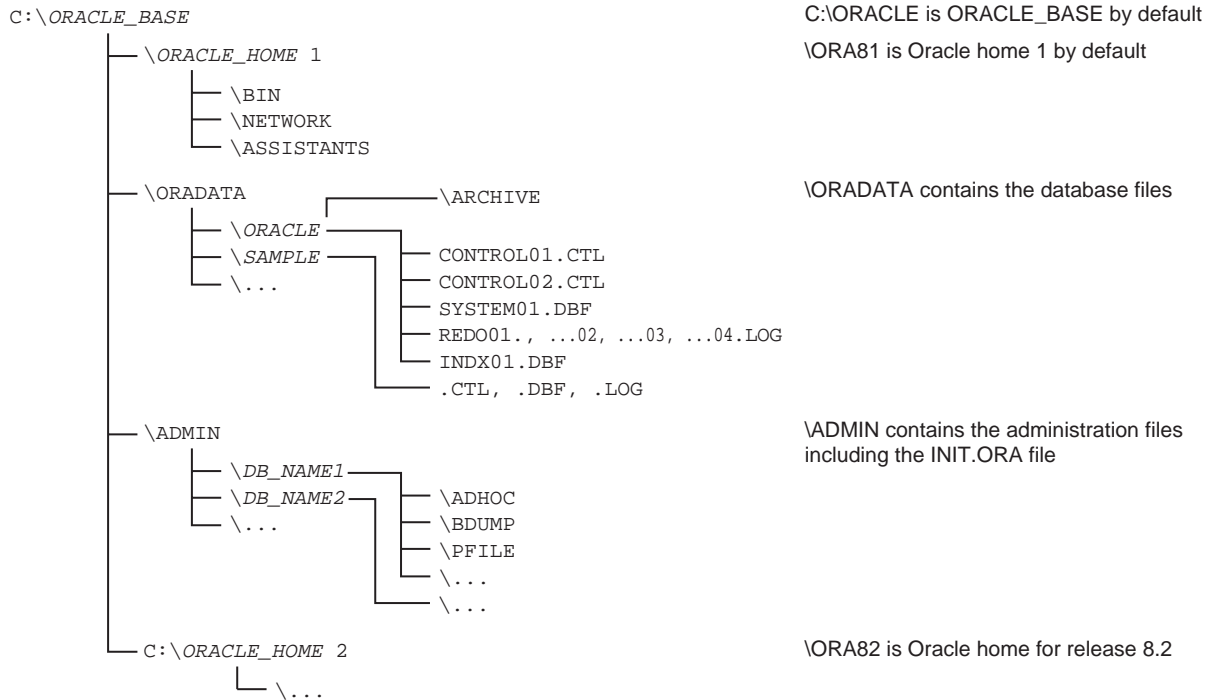
Directory Tree of a Sample OFA-Compliant Database

The following is the complete hierarchical directory tree of a sample OFA-compliant database:

X:\ORACLE_BASE			C:\ORACLE is the default ORACLE_BASE directory
	\ORACLE_HOME1		\ORA81 is the name of the first Oracle home by default
		\BIN	Subtree for Oracle binaries
		\NETWORK	Subtree for Net8
		\ASSISTANTS	Configuration assistants
	\ORADATA		Subtree for Oracle database files
		\DB_NAME1	Subtree for *DB_NAME1* database files
			CONTROL01.CTL Control file 1
			CONTROL02.CTL Control file 2
			CONTROL03.CTL Control file 3
			DRO1.DBF *inter*Media related objects
			SYSTEM01.DBF System tablespace data file
			RBS01.DBF Rollback tablespace data file
			INDX01.DBF Index tablespace data file
			TEMP01.DBF Temporary tablespace data file
			USERS01.DBF Users tablespace data file
			REDO01.LOG Redo log file group 1, member 1
			REDO02.LOG Redo log file group 2, member 1
			REDO03.LOG Redo log file group 3, member 1
		\DB_NAME2	Subtree for *DB_NAME2* database files
			CTL DBF LOG Control, data, and redo log files
		\DB_NAME3	Subtree for *DB_NAME3* database files
			CTL DBF LOG Control, data, and redo log files
	\ADMIN		Subtree for database administration files
		\DB_NAME1	Subtree for *DB_NAME1* database administration files
			\ADHOC Ad hoc SQL scripts
			\ADUMP Audit files
			\ARCH Archived redo log files
			\BDUMP Background process trace files
			\CDUMP Core dump files
			\CREATE Database creation files
			\EXP Database export files
			\PFILE Initialization parameter file
			\UDUMP User SQL trace files
		\DB_NAME2	Subtree for *DB_NAME2* database administration files
		. . .	
		\DB_NAME3	Subtree for *DB_NAME3* database administration files
		. . .	

ORACLE_HOME2	Second Oracle home
\\. . .	
ORACLE_HOME3	Third Oracle home
\\. . .	

> **Note:** The directory tree for multiple instance databases (Oracle Parallel Server installations) has additional subdirectories and files. See *Oracle Parallel Server Administrator's Guide* for more information.

OFA Directory Naming Conventions

OFA uses directory naming conventions that make it easy to identify the precise Oracle home and database name that is associated with a set of files. This section describes the naming conventions used for the top-level directories of an OFA-compliant database directory tree:

- ORACLE_BASE Directory

- ORACLE_HOME Directory

- ADMIN Directory

- ORADATA Directory

- DB_NAME Directory

ORACLE_BASE Directory

ORACLE_BASE is the root of the Oracle directory tree. If you install an OFA-compliant database using Oracle Universal Installer defaults, *ORACLE_BASE* is *X*:\ORACLE where *X* is any hard drive. For example, C:\ORACLE.

If you are installing Oracle8*i* for Windows NT on a clean computer, you may want to change *ORACLE_BASE* to an appropriate value before running Oracle Universal Installer. Most users will not need or want to do this.

Before you run Oracle Universal Installer for the first time, change the value of *ORACLE_BASE* at the system level. Only change the value of *ORACLE_BASE* before you run Oracle Universal Installer for the first time because if there is an existing *ORACLE_BASE*, and you change it, there will be a conflict of Oracle base directories. If you create another *ORACLE_BASE* when the original *ORACLE_BASE* already exists, certain tools and the database will not be able to find previously created files because they will look for them in the new *ORACLE_BASE* instead of the original *ORACLE_BASE*.

To change the value of *ORACLE_BASE* at the system level:

On Windows NT:

1. Choose Start > Settings > Control Panel.

   The *Control Panel* window appears.

2. Double-click the System icon.

   The *System Properties* window appears.

3. Click the *Environment* tab.

   The System Variables appear.

4. Type a new value for *ORACLE_BASE* in the Value text box, then click OK to exit.

On Windows 95 and Windows 98:

1. Open the AUTOEXEC.BAT file, using a text editor.

2. Edit the value of the ORACLE_BASE statement.

3. Reboot your computer.

> **Note:** An ORACLE_BASE registry key exists for every Oracle home. Ideally, the value of the ORACLE_BASE registry key will be identical for each Oracle home.

ORACLE_HOME Directory

ORACLE_HOME is located beneath *X:\ORACLE_BASE* and contains subdirectories for Oracle software executables and network files.

If you install Oracle8*i* for Windows NT on a clean computer and use the default settings, the first Oracle home directory that you create is called \ORA81.

ADMIN Directory

Database administration files are stored in subdirectories of *ORACLE_BASE* \ADMIN*DB_NAME*.

The following table describes the subdirectories for database administration files:

Subdirectories of \ADMIN*DB_NAME*	Contain...
\ADHOC	Ad hoc SQL scripts for a given database
\BDUMP	Background process trace files
\CDUMP	Core dump files
\CREATE	Database creation files
\EXP	Database export files
\PFILE	Initialization parameter files
\UDUMP	User process trace files

ORADATA Directory

Database files are stored in *ORACLE_BASE*\ORADATA*DB_NAME*.

The following table describes the database files:

Files in \ORADATA*DB_NAME*	Description
CONTROL01.CTL	Control file 1
CONTROL02.CTL	Control file 2
CONTROL02.CTL	Control file 3
OEMREP03.DBF	Oracle Enterprise Manager repository tablespace data file
SYSTEM01.DBF	SYSTEM tablespace data file
RBS01.DBF	RBS tablespace data file
INDX01.DBF	INDX tablespace data file
TEMP01.DBF	TEMP tablespace data file
USERS01.DBF	USERS tablespace data file
REDO01.LOG	Redo log file group one, member one
REDO02.LOG	Redo log file group two, member one
REDO03.LOG	Redo log file group three, member one

> **Note:** This directory structure only allows for disk striping on UNIX platforms. See "Support for Symbolic Links on Windows NT" on page 3-32.

DB_NAME Directory

DB_NAME is the unique name for a particular database and has the same value as the DB_NAME parameter in the initialization parameter file. When you create a database, DB_NAME can be no more than eight characters long and can contain only the following characters:

- Alphabetic characters
- Numbers
- Underscores (_)
- Pound sign (#)
- Dollar sign ($)

OFA and Multiple Oracle Home Configurations

The following sections describe various OFA and multiple Oracle home configurations.

Specifying an *ORACLE_HOME* Directory

To install an OFA-compliant database, you must specify an Oracle home directory in the *Path:* field of Oracle Universal Installer of the form:

X:\[*PATHNAME*]*ORACLE**HOME_NAME*

where:

X:\	is any hard drive. For example, C:\.
[*PATHNAME*]	is an optional directory pathname.
ORACLE	is a mandatory directory pathname unless you have changed the value of the ORACLE_BASE registry key before performing the installation. See "ORACLE_BASE Directory" on page 3-20 for information on how to change the *ORACLE_BASE* from the default value ORACLE.
HOME_NAME	is the name of the Oracle home.

The following are examples of OFA-compliant Oracle home directories:

- C:\TEST\ORACLE\ORA81
- D:\ORACLE\ORA81

Default OFA Database

To install a default OFA database:

1. Install Oracle8*i* for Windows NT release 8.1.6 on a clean computer (one with no other Oracle software on the computer), and accept the default Oracle Universal Installer settings for the first Oracle home (C:\ORACLE\ORA81) in the *Path:* field.

2. Complete the installation.

3. Run Oracle Universal Installer again and the same release a second time or release 8.2.*x* (when it is available). Accept the default Oracle Universal Installer settings for the first Oracle home (C:\ORACLE\ORA82) in the *Path:* field.

The default OFA database settings are as follows:

Setting	Value
ORACLE_BASE	is C:\ORACLE and is the same for all Oracle homes
Oracle home 1	is C:\ORACLE\ORA81
Oracle home 2	is C:\ORACLE\ORA82

This figure below illustrates the directory tree:

```
                              C:\ORACLE
                C:\ORACLE is ORACLE_BASE for both ORACLE homes

    \ADMIN              \ORA81              \ORADATA            \ORA82
                   C:\ORACLE\ORA81                        C:\ORACLE\ORA82
                   is Oracle home 1                       is Oracle home 2

                   \...   \NETWORK  \BIN              \...   \NETWORK  \BIN


\DB_NAME1  \DB_NAME2  DB_NAME3   \DB_NAME1      \DB_NAME2           \DB_NAME3
              \...       \...                      \...                \...

      ── \ADHOC                  ── CONTROL01.CTL
      ── \ARCH                   ── CONTROL02.CTL
      ── \BDUMP                  ── SYSTEM01.DBF
      ── \CDUMP                  ── RBS01.DBF
      ── \CREATE                 ── INDX01.DBF
      ── \EXP                    ── TEMP01.DBF
      ── \PFILE                  ── USERS01.DBF
      ── \UDUMP                  ── OEMREP01.DBF
                                 ── REDO01.LOG
                                 ── REDO02.LOG
                                 ── REDO03.LOG
                                 ── REDO04.LOG
```
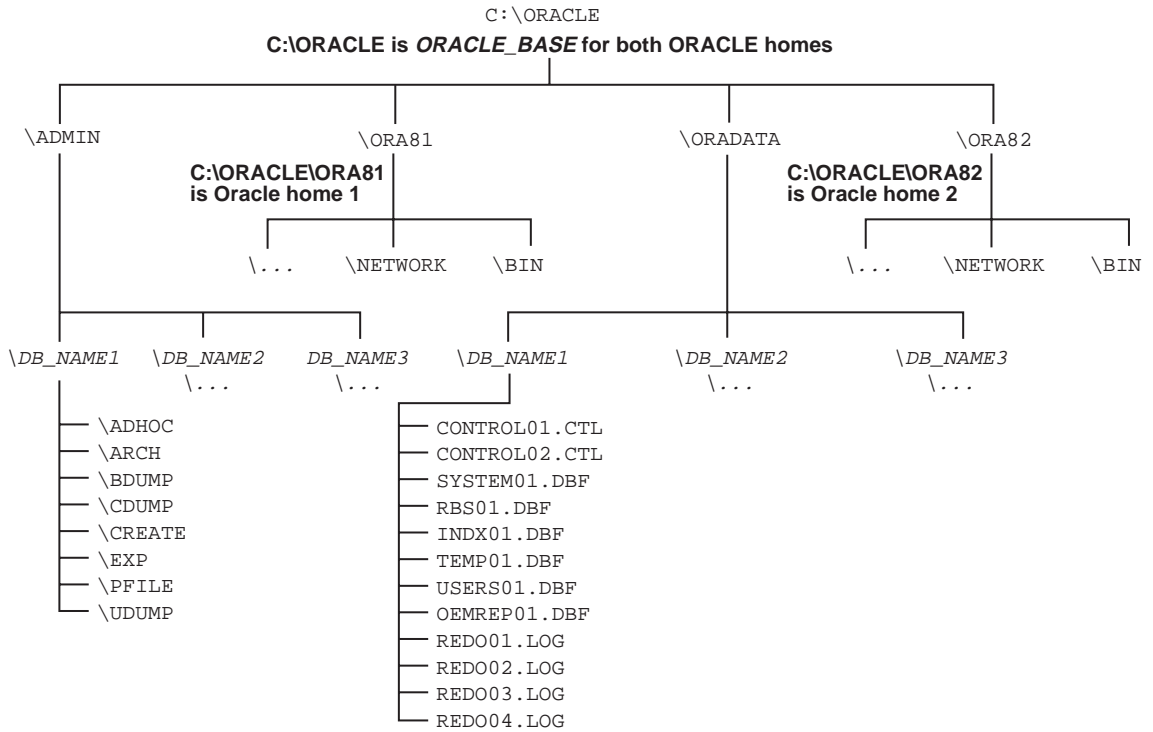
Non-Default OFA Database, Case 1

To install a non-default OFA database, case 1:

1. Install Oracle8*i* for Windows NT release 8.1.6 and change the default Oracle Universal Installer settings for the first Oracle home from C:\ORACLE\ORA81 in the *Path:* field to X:\XYZ.

2. Complete the installation.

3. Run Oracle Universal Installer again and change the default Oracle Universal Installer settings for the second Oracle home from C:\ORACLE\ORA82 in the *Path:* field to Y:\ABC.

For case 1, the non-default OFA database settings are as follows:

Setting	Value
ORACLE_BASE	is X:\XYZ for the first Oracle home and is Y:\ABC for the second Oracle home
Oracle home 1	is X:\XYZ
Oracle home 2	is Y:\ABC

This figure illustrates the resulting directory trees:

```
                                      X:\XYZ
          X:\XYZ is Oracle home 1 and ORACLE_BASE for Oracle home 1

     \ADMIN         \NETWORK                \BIN            \ORADATA

\DB_NAME1    \DB_NAME2    \DB_NAME3    \DBNAME1        \DBNAME2         \DBNAME3

   \...         \...         \...     .CTL .DBF .LOG  .CTL .DBF .LOG  .CTL .DBF .LOG
```

```
                                      Y:\ABC
          Y:\ABC is Oracle home 2 and ORACLE_BASE for Oracle home 2

     \ADMIN         \NETWORK                \BIN            \ORADATA

\DB_NAME1    \DB_NAME2    \DB_NAME3    \DBNAME1        \DBNAME2         \DBNAME3

   \...         \...         \...     .CTL .DBF .LOG  .CTL .DBF .LOG  .CTL .DBF .LOG
```
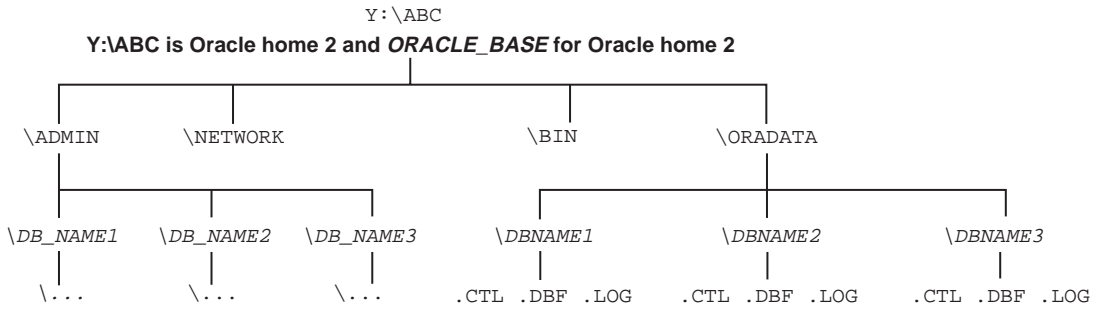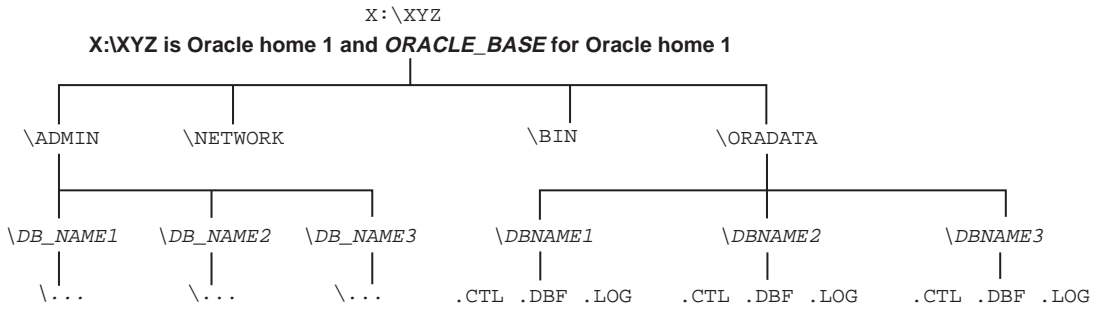
Non-Default OFA Database, Case 2

To install a non-default OFA database, case 2:

1. Install Oracle8*i* for Windows NT release 8.1.6 and change the default Oracle Universal Installer settings for the first Oracle home from C:\ORACLE\ORA81 in the *Path:* field to X:\XYZ\ORACLE\ABC.

2. Complete the installation.

3. Run Oracle Universal Installer again and change the default Oracle Universal Installer settings for the second Oracle home from C:\ORACLE\ORA82 to X:\PQR.

For case 2, the non-default OFA database settings are as follows:

Setting	Value
ORACLE_BASE	is X:\XYZ\ORACLE and is the same for both Oracle homes
Oracle home 1	is X:\XYZ\ORACLE\ABC
Oracle home 2	is X:\PQR

This figure illustrates the resulting directory tree:

```
                              X:\XYZ
                                │
                                │                           X:\PQR
                                │                       X:\PQR is Oracle home 2
                                ├───────────────────────────────┤
                                │                       ┌───────┴───────┐
                                │                    \NETWORK        \BIN
  X:\XYZ\ORACLE is
  ORACLE_BASE for
  both Oracle homes
                                │
        ┌───────────────────────┼───────────────────────────────────────────┐
   \ORACLE\ABC               \ADMIN                                      \ORADATA
X:\XYZ\ORACLE\ABC
 is Oracle home 1
   ┌─────┴─────┐        ┌───────┼───────┐                    ┌─────────────┼─────────────┐
\NETWORK   \BIN     \DB_NAME1  \DB_NAME2  \DB_NAME3       \DB_NAME1     \DB_NAME2     \DB_NAME3
                               \...       \...                           \...          \...
                         │                                    │
                         ├── \ADHOC                           ├── CONTROL01.CTL
                         ├── \ARCH                            ├── CONTROL02.CTL
                         ├── \BDUMP                           ├── SYSTEM01.DBF
                         ├── \CDUMP                           ├── RBS01.DBF
                         ├── \CREATE                          ├── INDX01.DBF
                         ├── \EXP                             ├── TEMP01.DBF
                         ├── \PFILE                           ├── USERS01.DBF
                         └── \UDUMP                           ├── OEMREP01.DBF
                                                              ├── REDO01.LOG
                                                              ├── REDO02.LOG
                                                              ├── REDO03.LOG
                                                              └── REDO04.LOG
```
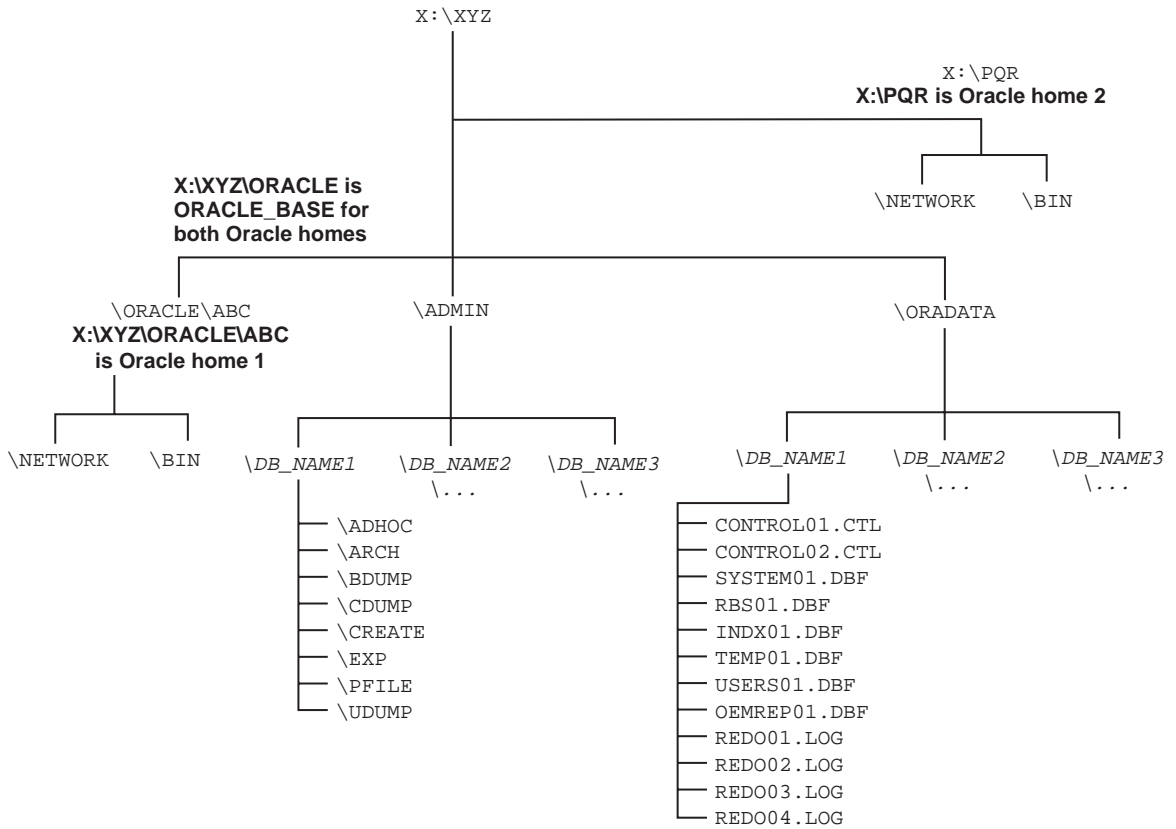
Increasing Reliability and Performance

One of the basic goals of OFA is to increase reliability and performance by distributing I/O load across different physical drives. If you are trying to maximize reliability and performance, Oracle Corporation recommends that you do the following:

- Read "Disk Mirroring" on page 3-30 and "Disk Striping" on page 3-30.

- Move your files to the disks on your system to take advantage of the recommendations

Disk Mirroring

Oracle log files and database files can be separated and treated with different levels of hardware reliability. Generally, Oracle log files are more highly reliable, because of redundancy. Creating reliability based on redundancy may require you to duplicate all of your data, using disk mirrors.

Disk mirroring can be done with the Windows NT Disk Administrator and commonly with hardware controllers. Two identical drives are usually required to construct a mirror, the concept being that if one disk fails, the other disk can be used to recover data that would otherwise be lost. Using one of the disks to recover lost data may involve "breaking" the mirror. If the mirror breaks, you need to build a new mirror.

You can achieve a lesser degree of redundancy by configuring the disks, using a Redundant Array of Inexpensive Disks (RAID) configuration provided by the disk controller. The RAID level determines the amount of redundancy. Some RAID levels may use the "hot swapping" feature. Hot swapping means that you can replace a bad disk with a good one without turning off the computer or losing functionality.

Disk Striping

How you set up disks for use in a database depends on the number of disks and the type of hard disk controllers available. If the hard disk controllers support both striping and mirroring, Oracle Corporation recommends you configure the controllers to support striping.

Some controllers are configured at system startup time by issuing a keyboard sequence that brings up configuration programs written by the controller manufacturer. One goal is to stripe as many drives together as possible by configuring the controllers. Each stripe shows up as one logical device.

Striping provides significant performance advantages. All the space from the striped drives appears as a single logical drive. Furthermore, the space is used by interlacing "stripes" of space from all of the disks in the stripe. This means that a large file uses some space from the first disk, then some from the second disk and so on to the last disk and then starting back at the first disk again. Each file may be spread over all of the striped disks. The data in such a file may be accessed randomly by more than one CPU without contention.

The controllers that support striping usually provide caching as well. This means that data may be written to the controller and cached and saved for a time in storage not on the disk. Data that is read can be cached on the controller in a similar fashion. Read caching is not necessary for Oracle databases since all database reads are cached already in the System Global Area (SGA). The value of the DB_BLOCK_ BUFFERS parameter in the initialization parameter file determines the number of buffers that can be used in the SGA. This value also configures the Oracle8*i* database on startup.

Using Raw Partitions for Tablespaces

A *raw partition* is a portion of a physical disk that is accessed at the lowest possible level. The I/O of a raw partition improves performance by approximately 5% to 10% compared to the I/O of a partition containing a file system. Therefore, Oracle Corporation encourages you to use raw partitions for your tablespaces.

> **See Also:** Appendix D, "Storing Tablespaces on Raw Partitions"

Comparison Between OFA on Windows NT and UNIX

You implement OFA on Windows NT and UNIX in the same way. However, differences exist with regard to the following:

- Directory Naming
- ORACLE_BASE Directory
- Support for Symbolic Links on Windows NT

> **See Also:** Your UNIX operating system-specific administrator's reference for information about OFA on UNIX.

Directory Naming

The top-level names of the OFA directory tree differ between Windows NT and UNIX. However, the main subdirectory and file names are the same on both operating systems.

ORACLE_BASE Directory

On Windows NT, *ORACLE_BASE* is associated with an Oracle home directory. *ORACLE_BASE* is defined in the registry (for example, in HKEY_LOCAL_ MACHINE \SOFTWARE\ORACLE\HOME0).

On UNIX, *ORACLE_BASE* is associated with a UNIX user's environment.

Support for Symbolic Links on Windows NT

Windows NT currently does not support symbolic links like those in UNIX, although Microsoft has announced the intention to support them in a near-future release.

The goal of OFA is to place all Oracle software under one *ORACLE_BASE* directory and to spread the files across different physical drives as your databases increase in size.

On Windows NT 4.0, this implies that everything is on the same hard drive, which may be neither feasible nor desirable.

On UNIX, although everything seems to be in one directory on the same hard drive, files can be on different hard drives if they are symbolically linked or have that directory as a mount point.

Oracle Corporation recommends that you use one logical drive to store your database administration files and that you place other files, as needed, on other logical drives in an ORADATA*DB_NAME* directory.

In the following example, there are four logical drives for a database named PROD:

- C:\ contains an Oracle home and the database administration files.

- F:\ contains the redo log files. (The F:\ drive could also represent two physical drives that have been striped to increase performance.)

- G:\ contains one of the control files and all of the tablespace files. (The G:\ drive could also use a RAID Level-5 configuration to increase reliability.)

- H:\ contains the second control file.

C:\ORACLE				First logical drive
	\ORA81			Oracle home
		\BIN		Subtree for Oracle binaries
		\NETWORK		Subtree for Net8
		\ . . .		
	\ADMIN			Subtree for database administration files
		\PROD		Subtree for PROD database administration files
			\ADHOC	Ad hoc SQL scripts
			\ADUMP	Audit files
			\BDUMP	Background process trace files
			\CDUMP	Core dump files
			\CREATE	Database creation files
			\EXP	Database export files
			\PFILE	Initialization parameter file
			\UDUMP	User SQL trace files
F:\ORACLE				Second logical drive (represents two physical drives that have been striped)
	\ORADATA			Subtree for Oracle database files
		\PROD		Subtree for PROD database files
			REDO01.LOG	Redo log file group one, member one
			REDO02.LOG	Redo log file group two, member one
			REDO03.LOG	Redo log file group three, member one
G:\ORACLE				Third logical drive (uses a RAID Level 5 configuration)
	\ORADATA			Subtree for Oracle database files
		\PROD		Subtree for PROD database files
			CONTROL01.CTL	Control file 1
			SYSTEM01.DBF	System tablespace data file
			RBS01.DBF	Rollback tablespace data file
			INDX01.DBF	Index tablespace data file
			TEMP01.DBF	Temporary tablespace data file
			USERS01.DBF	Users tablespace data file
H:\ORACLE				Fourth logical drive
	\ORADATA			Subtree for Oracle database files
		\PROD		Subtree for PROD database files
			CONTROL02.CTL	Control file 2

4

Using Oracle8*i* Directory Server Features with Active Directory

This chapter describes how to enable Oracle8*i* directory server features with Microsoft's Active Directory.

Specific topics discussed are:

- Overview
- Oracle8i Directory Server Features
- Integration with Active Directory
- Requirements for Using Oracle8i with Active Directory
- Installing and Configuring Oracle8i in an Active Directory Environment
- Testing Connectivity
- Managing Access Control Lists for Oracle Directory Objects
- Creating Security Domains

Overview

This section provides an overview of the following topics:

- What are LDAP and a Directory Server?
- What is Active Directory?

What are LDAP and a Directory Server?

The Lightweight Directory Access Protocol (LDAP) is a networking and directory access protocol for accessing information in a directory server. The directory server centrally stores and manages information about all network resources and makes that information accessible to users and applications. Resources can include user names, databases, computers, fax servers, applications, e-mail addresses, and printers. A directory server is analogous to a telephone directory, which stores information such as phone numbers and addresses of telephone subscribers.

What is Active Directory?

Active Directory is the LDAP-compliant directory server included with Windows 2000. Active Directory centrally stores all Windows 2000 information, including users, groups, and policies. Active Directory also stores information about network resources such as databases, and makes this information available to application users and network administrators. Active Directory enables users to access network resources with a single login. The scope of Active Directory can range from storing all the resources of a small computer network to storing all the resources of several wide areas networks (WANs).

Oracle8*i* Directory Server Features

With Oracle8*i* release 8.1.6, two new features are provided for storing Oracle information in a directory server. These new features are briefly described in the following sections:

- Net8 Directory Naming Features
- Enterprise User Security Features

References are provided to additional documentation. Both features have been enabled to work with Microsoft's Active Directory.

Net8 Directory Naming Features

This feature enables you to create and store database service and net service name entries for use with Net8 as directory objects in Active Directory. These objects contain connectivity information that can be used by various Oracle client applications when connecting to an Oracle8*i* database.

During Oracle8*i* release 8.1.6 database creation, a database service entry is created with Oracle Database Configuration Assistant. Clients configured to access the directory server can use this entry in their connect strings to connect to the database without any additional configuration.

If you prefer not to expose the database service entry to clients, you can use Net8 Assistant to create net service name entries in the directory server, which eliminates the need to create and maintain separate TNSNAMES.ORA files on each client computer. When clients attempt an Oracle8*i* database connection, the net service name is instead retrieved from a directory server. The Directory Server Migration Wizard, available with Net8 Assistant, enables you to export net service names stored in an existing TNSNAMES.ORA file to the directory server.

> **Note:** Database service and net service name entries stored in an Oracle Names server can migrated to a directory server using the NAMESCTL utility. See the *Net8 Administrator's Guide* for more information.

This chapter frequently references Net8 directory naming terms and concepts. Read the following documentation for descriptions of terms and concepts that an administrator and client user must understand before using an Oracle8*i* database with Active Directory.

See Section...	Which Describes...
"Net8 and an LDAP-Compliant Directory Server" in Chapter 2 of the *Net8 Administrator's Guide*	■ How client computers use a directory server to connect to an Oracle8*i* database ■ Naming conventions and the location for Net8 and Oracle8*i* database entries in a directory server ■ How Oracle8*i* database service and net service name entries are created or modified ■ Database server requirements for using a directory server for lookups ■ Client requirements for performing entry lookups in a directory server ■ How to create connect strings to connect to an Oracle8*i* database using database service and net service names stored in a directory ■ Access Control List security overview (the client's privileges in the directory)
"Configuring the Directory Naming Method" in Chapter 6 of the *Net8 Administrator's Guide*	■ The Oracle8*i* and Active Directory configuration process (at the end of or separate from server and client installation) ■ Adding and removing users from the OracleNetAdmins group with the LDAPMODIFY utility ■ Exporting information into Active Directory from an existing TNSNAMES.ORA file or Oracle Names server

Enterprise User Security Features

This feature enables you to create and store Oracle8*i* database information as directory objects in Active Directory. This enables users to make natively-authenticated, Single Sign-On (SSO) connections to a database. An administrator can create and store enterprise users and roles for the Oracle8*i* database in Active Directory, which helps centralize the administration of users and roles across multiple databases.

This chapter frequently references enterprise user security terms and concepts. Read the following documentation for descriptions of terms and concepts that an administrator and client user must understand before using an Oracle8*i* database with Active Directory.

See...	Which Describes...
Chapter 17, "Managing Enterprise User Security" of *Oracle Advanced Security Administrator's Guide*	■ Enterprise user security and management ■ Descriptions of enterprise users, roles, domains, and concepts ■ Location for enterprise user security entries in a directory server ■ Installing and configuring enterprise user security
Chapter 20, "Using Oracle Enterprise Security Manager" of *Oracle Advanced Security Administrator's Guide*	■ Creating and managing enterprise users, roles, and domains

> **Note:** Oracle Enterprise Security Manager cannot create or delete Windows 2000, Windows NT, Windows 95, or Windows 98 operating system user names. Instead, Oracle Enterprise Security Manager creates a contact name in Active Directory. You cannot log in with a contact name; it is just defined for external purposes. You can then assign roles to this "user". You then assign this contact user name to a global user.

> **Note:** Enterprise domains are directory constructs consisting of Oracle8*i* databases and enterprise users and roles. Enterprise domains are different from Windows 2000 domains, which are a collection of computers that share a common directory database.

Integration with Active Directory

In addition to Net8 directory naming and enterprise user security integration with a directory server, the following features have been specifically integrated into Active Directory:

- Automatic Discovery of Directory Servers
- Integration with Microsoft Tools
- User Interface Extensions for Net8 Directory Naming
- Enhancement of Directory Object Type Descriptions
- Integration with Windows Login Credentials
- How Do Oracle Directory Objects Display in Active Directory?

Automatic Discovery of Directory Servers

Net8 Configuration Assistant enables you to configure client computer and Oracle8*i* database server access to a directory server. When Net8 Configuration Assistant starts at the end of Oracle8*i* database installation or is manually started after installation, it prompts you to specify a directory server type to use. When you select Active Directory as your directory server type, Net8 Configuration Assistant automatically:

- Discovers the Active Directory server location
- Configures access to the Active Directory server
- Creates the administrative context (also known as your domain)

If the Active Directory server through which client connections are accessing an Oracle8*i* database is shut down, another Active Directory server is automatically discovered and begins providing connection information; this prevents any downtime for client connections.

You must be running your Oracle client and database software in a Windows 2000 domain to take advantage of the automatic directory server discovery features of Net8 Configuration Assistant. This is regardless of the Oracle client and database releases you are using.

If you are not running in a Windows 2000 domain, Net8 Configuration Assistant does not automatically discover your directory server, and instead prompts you for additional information, such as the naming context and Active Directory location.

Integration with Microsoft Tools

Oracle8*i* database service, Net8 net service name, and enterprise role entries in
Active Directory display in the following Microsoft Windows 2000 tools:

Tool	Description	This Integration Enables You To...
Windows Explorer	A user tool that displays the hierarchical structure of files, directories, and local and network drives on your computer.	Display and test Oracle8*i* database service and Net8 net service name objects
Active Directory Users and Computers	An administrative tool installed on Windows servers configured as domain controllers. This tool enables you to add, modify, delete, and organize Windows 2000 accounts and groups, and publish resources in your organization's directory.	Display and test Oracle8*i* database service and Net8 net service name objects and manage access control

See Also:

- "Testing Connectivity from Microsoft Tools" on page 4-17
- "Managing Access Control Lists for Oracle Directory Objects" on page 4-20

## User Interface Extensions for Net8 Directory Naming

The property menus of Oracle8*i* database service and net service name objects in
Windows Explorer and Active Directory Users and Computers have been enhanced.
These enhancements enable you to test for object connectivity to the Oracle8*i*
database and perform database administration. When you right click these Oracle
directory objects, a menu presents you with two options for testing connectivity:

Menu Option	Description
Test	Starts an application that tests that the user name, password, and net service name you initially entered can connect to the Oracle8*i* database.
Connect with SQL*Plus	Starts SQL*Plus, which enables you to perform database administration, run scripts, and so on.

See Also: "Testing Connectivity from Microsoft Tools" on
page 4-17 for more information

Enhancement of Directory Object Type Descriptions

Oracle directory object type descriptions in Active Directory have been enhanced to make them easier to understand. For example, here is the description for OracleDefaultDomain's type in the Type column of the right window pane:

enhanced object type descriptions
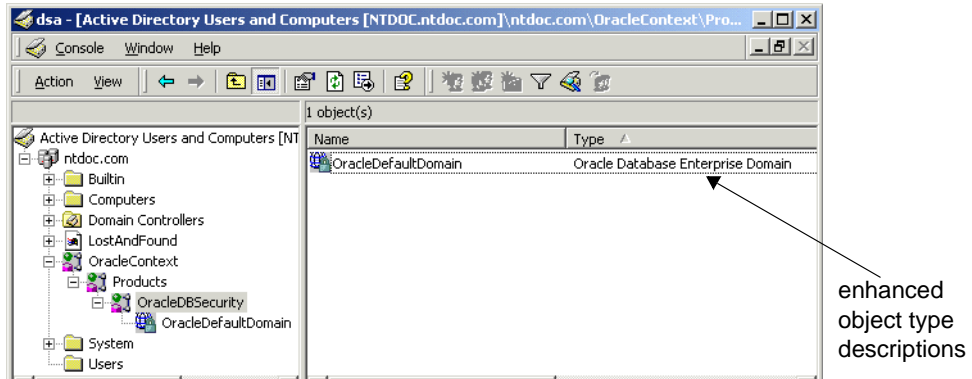
Integration with Windows Login Credentials

SSO enables client users to access all authorized network resources (such as Active Directory) with a single authentication that is performed when they initially specify their user login credentials to access the network. SSO is included in Windows 2000 through the Kerberos and Secure Sockets Layer (SSL) authentication protocols.
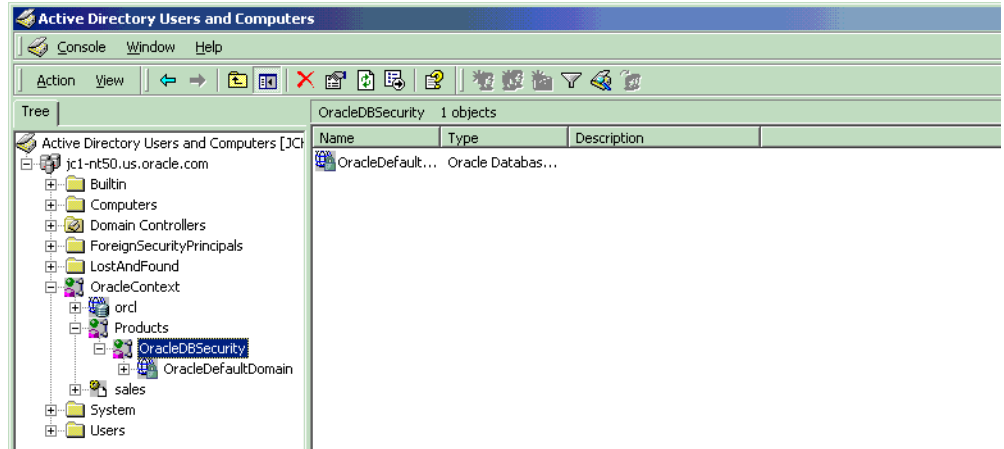
The Oracle8*i* database and configuration tools can use the Windows user's login credentials to automatically connect to Active Directory without having to re-enter their login credentials. This enables:

- Oracle8*i* clients and databases to securely connect to Active Directory and retrieve net service name, enterprise user, and enterprise role information

- Configuration tools such as Oracle Enterprise Security Manager, Net8 Configuration Assistant, Net8 Assistant, and Oracle Database Configuration Assistant to connect automatically to Active Directory and configure the Oracle8*i* database and net service name objects

For Windows 2000, the default authentication protocol used is Kerberos.

How Do Oracle Directory Objects Display in Active Directory?

When the Oracle8*i* database and Net8 are installed and configured to access Active Directory, Oracle directory objects appear in Active Directory Users and Computers:

This table describes these Oracle directory objects:

Object	Description
domain	The domain (also known as the administrative context) in which you created your Oracle Context. The administrative context contains various Oracle entries to support directory naming and enterprise user security. Net8 Configuration Assistant automatically discovers this information during Oracle8*i* database integration with Active Directory.
OracleContext	The top-level Oracle entry in the Active Directory tree that can contain Oracle8*i* database service and Net8 net service name object information. All Oracle software information is placed in this container.
orcl	The Oracle8*i* database service name (for this example, *orcl* is the name).
Products	A container for Oracle security and domain information.
OracleDBSecurity	A container for security domains.
OracleDefaultDomain	The default enterprise domain created. You can create additional enterprise domains with Oracle Enterprise Security Manager.
sales	The net service name object (for this example, *sales* is the name).
Users	The folder for the three Oracle security groups. See section "Managing Access Control Lists for Oracle Directory Objects" on page 4-20 for more information. Enterprise users and roles created with Oracle Enterprise Security Manager also appear in this folder.

Requirements for Using Oracle8*i* with Active Directory

The requirements that you must complete depend upon the Oracle features you want to use:

	Required For...	
Requirement	**Net8 Directory Naming?**	**Enterprise User Security?**
"Oracle Schema Creation Requirements" on page 4-10	Yes	Yes
"Oracle Context Creation Requirements" on page 4-11	Yes	Yes
"Net8 Directory Naming Requirements" on page 4-12	Yes	No
"Enterprise User Security Requirements" on page 4-13	No	Yes

> **Note:** The Oracle schema and Oracle Context are both created when you run Net8 Configuration Assistant.

> **Note:** You must be running your Oracle clients and database server in a Windows 2000 domain. This is regardless of the Oracle client and Oracle database server releases you are running.

Oracle Schema Creation Requirements

Complete the following Oracle schema creation requirements to use the Net8 directory naming and enterprise user security features with Active Directory. A schema is a set of rules for Net8 and Oracle8*i* database entries and their attributes stored in Active Directory.

- You can create only one Oracle schema per forest.

- Perform schema creation on a Windows 2000 domain controller.

- The Windows 2000 domain controller must be the operations master. See your Microsoft documentation for instructions.

- Log in as a member of the Schema Administrator group to create the schema. Domain administrators by default are in the Schema Administrator group.

- Create the *Schema Update Allowed* registry parameter prior to installation. See your Microsoft documentation for instructions.

- Use Net8 Configuration Assistant to create the Oracle schema. You can create your schema during or after installation. The schema can be created by running Net8 Configuration Assistant on the Oracle8*i* database or on a client computer.

  > **See Also:** *Net8 Administrator's Guide* for configuration procedures and *Oracle8i Installation Guide for Windows NT* for a configuration overview

Oracle Context Creation Requirements

You must complete the following Oracle Context creation requirements to use the Net8 directory naming and enterprise user security features with Active Directory. The Oracle Context is the top-level Oracle entry in the Active Directory tree that contains Oracle8*i* database service and Net8 net service name object information.

- You can create only one Oracle Context per Windows 2000 domain (administrative context).

- You must have the right to create domain objects in order to create the Oracle Context in Active Directory with Net8 Configuration Assistant. If you are a domain administrator, you automatically have these rights.

- Use Net8 Configuration Assistant to create your Oracle Context. You can create the Oracle Context during or after installation.

  > **See Also:** See the *Oracle8i Installation Guide for Windows NT* for installation procedures and the *Net8 Administrator's Guide* for configuration procedures

Net8 Directory Naming Requirements

Ensure that you first satisfy the requirements described in:

- "Oracle Schema Creation Requirements" on page 4-10

- "Oracle Context Creation Requirements" on page 4-11

This table describes the Microsoft and Oracle software releases that must be installed to use Net8 directory naming with Active Directory:

For...	The Required Microsoft Software Is...		The Required Oracle Software Is...
Client Computers	■	Windows 2000	Oracle8*i* Client release 8.1.6, which includes Net8 Client and these configuration tools:
	■	Windows 4.0 with Active Directory Service Interfaces (ADSI)	■ Net8 Configuration Assistant
	■	Windows 95 or 98 with the Distributed Systems Client upgrade	■ Net8 Assistant
			Note: See the *Oracle8i Installation Guide for Windows NT* for installation instructions and "Required Configuration Tools" on page 4-14 for descriptions of the tasks that these configuration tools perform.
Database Server	■	Windows NT 4.0	Oracle8*i* database release 8.1.6 is required for registering the database service as an object in Active Directory.
	■	Windows 2000	

Enterprise User Security Requirements

Ensure that you first satisfy the requirements described in:

- "Oracle Schema Creation Requirements" on page 4-10

- "Oracle Context Creation Requirements" on page 4-11

This table describes the Microsoft and Oracle software releases required to use enterprise user security with Active Directory:

For...	Microsoft Software	Oracle Software
Client Computers	- Windows 2000 - Windows NT 4.0 with *or* without ADSI - Windows 95 or 98 with *or* without the Distributed Systems Client upgrade	Oracle8*i* Client release 8.1.5 or greater

Database Server	- Windows NT 4.0 with ADSI
	- Windows 2000
		Oracle database release 8.1.6, which automatically installs:
		- Oracle Database Configuration Assistant
		- Net8 Client, which includes these configuration tools:
		  - Net8 Configuration Assistant
		  - Net8 Assistant

Remote computer from which to manage the Oracle8*i* database		Oracle Enterprise Manager Console release 2.1, which includes: - Oracle Enterprise Security Manager - Net8 Client **Note:** Oracle Enterprise Security Manager is required if you want to create and manage enterprise users, roles, and domains.

Installing and Configuring Oracle8*i* in an Active Directory Environment

This section provides an overview of installation and configuration information. Specific topics covered include:

- Installation Tasks
- Required Configuration Tools
- Post-Installation Configuration Tasks

Installation Tasks

See Chapters 4 and 5 of the *Oracle8i Installation Guide for Windows NT* for Oracle8*i* installation instructions.

Required Configuration Tools

Several tools are required for configuring the Oracle clients and Oracle8*i* database for access to Active Directory. This table identifies:

- Tasks that these tools perform
- How to run these tools
- Documentation for additional information on using these tools

These tools are listed in the order in which to use them. After you configure your environment, you can take advantage of the Net8 directory naming and enterprise user security features.

To...	Run this Tool...	When Does This Tool Run?	For More Information...
Create an Oracle schema and Oracle Context in Active Directory (if one is not already installed) and Set up Access Control Lists for security in Active Directory	Net8 Configuration Assistant, which guides you through Oracle8*i* database server configuration with Active Directory. Run this tool either on the Oracle8*i* database server or from a client computer that connects to the server.	There are two methods: ■ Automatically started at the end of Custom installation of Oracle8*i* Server (the Oracle8*i* database) ■ Manually started after Oracle8*i* Server	■ Chapter 6 of *Net8 Administrator's Guide* ■ Chapters 4 and 5 of *Oracle8i Installation Guide for Windows NT* ■ online help included with Net8 Configuration Assistant

To...	Run this Tool...	When Does This Tool Run?	For More Information...
Register the Oracle8*i* database as an object in Active Directory **Note:** This task is not required if you are not using the enterprise user security feature.	Oracle Database Configuration Assistant	There are two methods: ■ Automatically started after Net8 Configuration Assistant has created the Oracle schema, Oracle Context, and set up Access Control List security, *if you* installed the Oracle8*i* database through the Custom installation type. ■ Manually started after Oracle8*i* Server installation and Active Directory access configuration by Net8 Configuration Assistant. Select the Change database configurations option.	■ *Oracle8i Administrator's Guide for Windows NT* ■ Chapters 4 and 5 of *Oracle8i Installation Guide for Windows NT*
Configure an Oracle8*i* client computer to access Active Directory	Net8 Configuration Assistant, which guides you through client computer configuration with Active Directory by prompting you to: ■ Select the Directory Naming Method as the naming method with which to connect to the Oracle8*i* database ■ Identify the Active Directory with which to integrate the Oracle client	There are two methods for running Net8 Configuration Assistant: ■ Automatically started at the end of any Oracle8*i* Client installation type: (Administrator, Application User, Programmer, or a Custom installation of Net8 Client) ■ Manually started after installation of any Oracle8*i* Client installation type	■ Chapter 6 of *Net8 Administrator's Guide* ■ Chapters 4 and 5 of *Oracle8i Installation Guide for Windows NT* ■ online help included with Net8 Configuration Assistant
Create and modify net service name objects or modify Net8 attributes of the database	Net8 Assistant	You must manually start Net8 Assistant.	■ *Net8 Administrator's Guide* ■ online help included with Net8 Assistant
Create enterprise users, roles, and domains in Active Directory (enterprise user security)	Oracle Enterprise Security Manager	Manually started as an integrated application of Oracle Enterprise Manager Console	■ *Oracle Advanced Security Administrator's Guide*

> **Note:** Oracle Enterprise Security Manager is a feature of Oracle
> Advanced Security and can only be used if you have purchased an
> Oracle Advanced Security license.

Post-Installation Configuration Tasks

You must set the OSAUTH_X509_NAME registry parameter to TRUE to use
enterprise user security. See "Enterprise User Authentication" on page 8-8 for more
information.

Testing Connectivity

This section describes how to connect to an Oracle8*i* database through Active
Directory. Specific topics discussed include:

- Testing Connectivity from Client Computers
- Testing Connectivity from Microsoft Tools

Testing Connectivity from Client Computers

Client computers connect to an Oracle8*i* database by specifying the database entry
that appears in the Oracle Context. For example, if the database service entry under
the Oracle Context in Active Directory was *sales*, a user connects through SQL*Plus
to the Oracle8*i* database as follows:

If the Client and Oracle8*i* database are in...	The Client Specifies The Following...
The same domain	`SQL> CONNECT SCOTT/TIGER@SALES`
Different domains	`SQL> CONNECT SCOTT/TIGER@SALES@DOMAIN` where *domain* is the domain in which the Oracle8*i* database is located.

The connect strings in this table follow DNS-style conventions. While Active
Directory also supports connections using X.500 naming conventions, DNS-style
conventions are the recommended method because of ease of use. DNS-style
conventions enable client users to access an Oracle8*i* database through a directory
server by entering minimal connection information; this is the case even when the
client computer and Oracle8*i* database are in separate domains. X.500 names are
longer; this is especially the case when the client and Oracle8*i* database are located
in different domains (also known as administrative contexts).

To learn more about X.500 naming conventions, see Chapter 2, "Net8 Concepts", of the *Net8 Administrator's Guide* for information.

Testing Connectivity from Microsoft Tools

Oracle directory objects in Active Directory are integrated with Microsoft tools such as:

- Windows NT Explorer
- Active Directory Users and Computers

You can perform the following tasks from within these Microsoft tools:

- Connect with SQL*Plus to an Oracle8*i* database
- Test Oracle8*i* database connectivity

> **Note:**   All clients accessing an Oracle8*i* database through Active Directory require read access on all Net8 net service name objects in the Oracle Context and must be able to authenticate anonymously with Active Directory. Net8 Configuration Assistant automatically sets this up.

Accessing Connectivity Tools

To access connectivity tools:

1. Start the Microsoft tool with which you want to connect:

With...	Choose...
Active Directory Users and Computers	1. Start > Programs > Administrative Tools > Active Directory Users and Computers.
Windows Explorer	1. Start > Programs > Accessories > Windows Explorer
	2. Expand My Network Places.
	3. Expand Entire Network.
	4. Expand Directory.

2. Expand the domain in which your Oracle Context is located.
3. Go to your Oracle Context.
4. Right click the object that defines your net service name attributes.

A menu appears with several options:
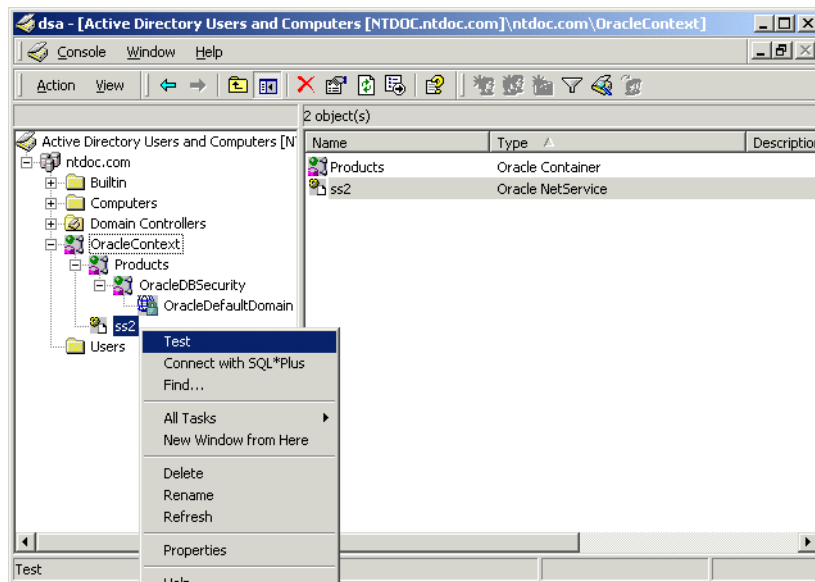
5. Make an appropriate selection:

If You Want To...	Then...	
Test connectivity	**1.**	Choose Test.
	2.	Go to section "Testing Connectivity" on page 4-19
Connect with SQL*Plus	**1.**	Choose Connect with SQL*Plus.
	2.	Go to section "Connecting With SQL*Plus" on page 4-19

Testing Connectivity

A status message appears describing the status of your connection attempt:
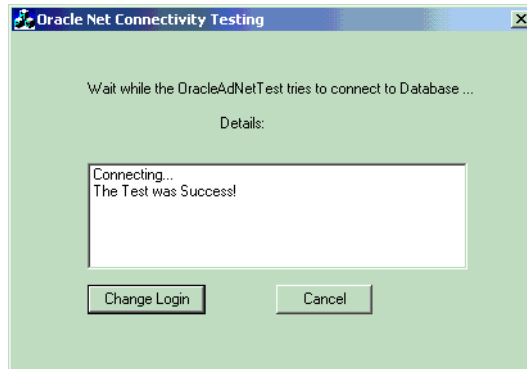
Connecting With SQL*Plus

The *Oracle SQL*Plus Login* dialog box appears:
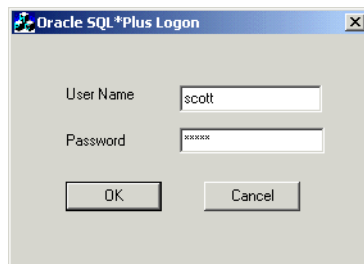
1. Enter your user name and password.

A status message appears describing the status of your connection attempt.

Managing Access Control Lists for Oracle Directory Objects

Access Control Lists provide Active Directory security by specifying:

- The user that can access the object attributes in the object

- Authentication method to access the entry

- Access rights, or what the user can do with the object (read/write) attributes in the object

Three security groups are automatically created when the Oracle Context is created in Active Directory. The user configuring access (and thus creating the Oracle Context) is automatically added to each:

Group	Description
OracleDBSecurityAdmin	Group for the creator of the Oracle Context. Users in this group can also: ■ Manage the group membership for all three security groups ■ Manage any object in the Oracle Context ■ Use Oracle Enterprise Security Manager to create security domains
OracleDBCreator	Group for the creator of the Oracle8*i* database. Users in this group can: ■ Modify the Oracle8*i* database objects that they create ■ Read, but not modify, the membership for this group The domain administrator is automatically a member of this group. **Note:** The domain administrator can start Oracle Database Configuration Assistant and Net8 Assistant, but cannot use Oracle Enterprise Security Manager. You must be a member of the OracleDBSecurityAdmins group to use Oracle Enterprise Security Manager.
OracleNetAdmins	Users in this group can: ■ Create and modify net service name objects ■ Read the group membership of this group ■ Modify the net service information in the database objects

Accessing the Security Groups

Active Directory Users and Computers enables you to add or remove users or change permission settings in the three security groups.

There are several tools available for adding or removing users:

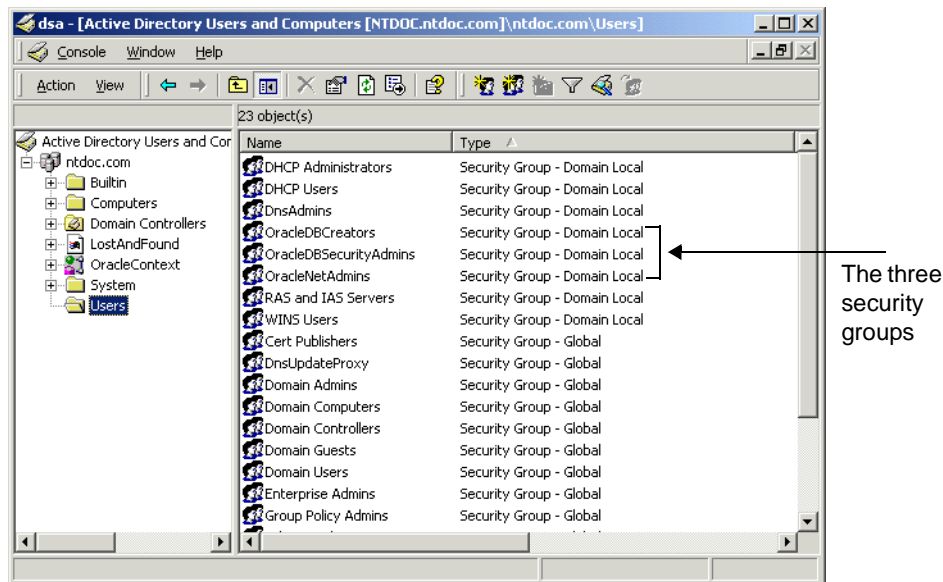| If You Want to...	Use...
Add or remove users in OracleNetAdmins	Active Directory Users and Computers
Add or remove users in OracleDBSecurityAdmin or OracleDBCreator	Oracle Enterprise Security Manager or Active Directory Users and Computers

This section describes how to use Active Directory Users and Computers. See the *Oracle Advanced Security Administrator's Guide* for instructions on using Oracle Enterprise Security Manager.

> **Note:** Use Active Directory Users and Computers to perform the procedures described in this section. Windows Explorer does not provide the functionality.

To add or remove users or change permission settings:

1. Choose Start > Programs > Administrative Tools > Active Directory Users and Computers.

2. Choose Advanced Features from the View main menu.

   This enables you to view and edit information that is normally hidden.

3. Expand the domain (administrative context) in which your Oracle Context is located.

4. Expand Users.

The three security groups appear in the right window pane:

5. Right-click the Oracle security group that you want to view or modify.

   A menu appears with several options.

6. Choose Properties.
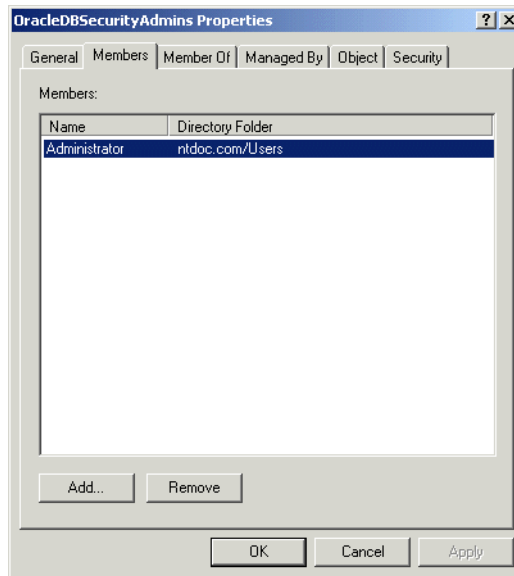
7. Make an appropriate selection:

If You Want To...	Then...	
Add or remove users	1.	Click the Members tab.
	2.	Go to section "Adding or Removing Users" on page 4-23.
Change permissions	1.	Click the Security tab.
	2.	Go to section "Changing User Permissions" on page 4-24.

Adding or Removing Users

To add or remove users:

1. Complete the access procedures in "Accessing the Security Groups" on page 4-21.

   The *Properties* dialog box for the group you selected appears (in this example, *OracleDBSecurityAdmins*):

2. Make an appropriate selection:

To...	Then...	
Add Users	1.	Click Add.
		The *Select Users, Contacts, Computers, or Groups* dialog box appears.
	2.	Select appropriate users or groups, and click Add.
		Your selections appear in the *Select Users, Contacts, Computers, or Groups* dialog box.
	3.	Click OK.

| To...	Then...
Remove Users	**1.** Select a user to remove.
	2. Click Remove.
	The user is removed.
	3. Click OK.

Changing User Permissions

To change user permissions:

1. Complete the access procedures in "Accessing the Security Groups" on page 4-21.

The *Properties* dialog box for the group you selected appears.

2. Click Advanced.

3. Click View/Edit.

The *Permission Entry* dialog box for the security group you selected appears:
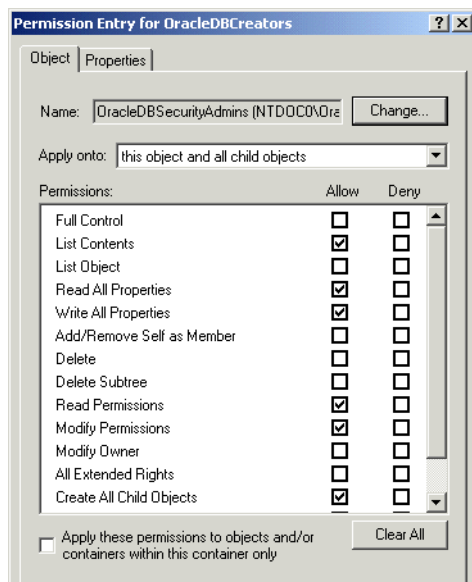
4. View or make appropriate changes to group permissions.

5. Click OK.

Creating Security Domains

A default security domain, OracleDefaultDomain, is created in your Oracle Context. If you do not want to use this domain or want to create another domain, use Oracle Enterprise Security Manager to create additional security domains (called enterprise domains). These domains are added under the OracleDBSecurity folder.

5

Post-Installation Configuration Tasks

This chapter describes some of the configuration tasks you must perform before using products like Oracle *inter*Media and the Oracle options. Where appropriate, the chapter provides references to other guides for those configuration tasks.

Specific topics discussed are:

- Oracle interMedia
- Oracle Spatial
- Oracle Time Series
- Oracle Visual Information Retrieval
- Multithreaded Server Support
- Advanced Replication

> **Note:** The directory path examples in this chapter follow Optimal Flexible Architecture (OFA) guidelines (for example, *ORACLE_BASE**ORACLE_HOME*\RDBMS\ADMIN). If you specified non-OFA compliant directories during installation, your directory paths will differ. See "OFA and Multiple Oracle Home Configurations" on page 3-23 for more information.

Oracle *inter*Media

Oracle enables file management in a variety of media, from text, to audio, to video. Files in each medium are managed through a specific *option* such as Oracle *inter*Media. This section describes *inter*Media configuration and describes each medium management component. Oracle *inter*Media is available with Oracle8*i* Enterprise Edition only.

Audio

Oracle *inter*Media Audio manages audio data in multiple file formats in an Oracle database. Types of audio data supported include conversations, songs, and other sounds in popular audio file formats. This makes it possible to integrate audio data with other application-specific object-relational data.

Video

Oracle *inter*Media Video manages video data in multiple video file formats. This makes it possible to integrate video data with other application-specific object-relational data.

Image

Oracle *inter*Media Image provides image storage, retrieval, and format conversion capabilities through an object data type (ODT). It also supports image storage, using Binary Large Objects (BLOBs), and references to image data residing in external files (BFILEs).

The Image component of Oracle *inter*Media also comes with a sample demonstration that shows how an image is extracted from an Oracle database.

Locator

Oracle *inter*Media Locator enables Oracle8*i* to support online internet-based geocoding facilities for locator applications and proximity queries.

Text

Oracle *inter*Media Text (formerly called ConText) enables text queries through SQL and PL/SQL from most Oracle interfaces.

By installing Oracle *inter*Media Text with an Oracle database server, client tools such as SQL*Plus, Oracle Forms, and Pro*C/C++ are able to retrieve and manipulate text in an Oracle database.

Oracle *inter*Media Text manages textual data in concert with traditional data types in an Oracle database. When text is inserted, updated, or deleted, Oracle *inter*Media Text automatically manages the change.

*Oracle inter*Media Audio, Video, Image, and Locator Configuration Responsibilities

Review the following table to determine how to configure Oracle *inter*Media Audio, Video, Image, and Locator.

If You...	Then...
Installed Oracle *inter*Media through the Oracle8*i* Enterprise Edition Typical installation type path	No manual configuration is required. All tasks described in "Configuring Oracle interMedia Audio, Video, Image, and Locator" on page 5-4 are automatically performed.
Installed both Oracle *inter*Media and Oracle8*i* Server together through the Oracle8*i* Enterprise Edition Custom installation type path	Oracle Database Configuration Assistant starts at the end of installation. If you select either of the following options: ■ Custom ■ Typical, and then select the Create new database files suboption Oracle Database Configuration Assistant asks if you want Oracle *inter*Media to be automatically configured.
Installed Oracle *inter*Media during a *separate* installation from Oracle8*i* Enterprise Edition	You must manually configure Oracle *inter*Media by either: ■ Performing the tasks under "Configuring Oracle interMedia Audio, Video, Image, and Locator" on page 5-4. ■ Starting Oracle Database Configuration Assistant and selecting Modify a database. Then following the on-screen instructions.
Manually copy your Oracle7 LISTENER.ORA and TNSNAMES.ORA files into your Oracle8*i* network directory	You must modify the TNSNAMES.ORA and LISTENER.ORA network configuration files on your server to enable external procedure calls to work and interMedia to function properly. Follow the tasks in "Configuring Net8 for External Procedures" in Chapter 8 of *Net8 Administrator's Guide.*
Want to use the demos	Perform the tasks described for the configuration of Oracle *inter*Media sample demos below.
Want to build the Oracle *inter*Media Image sample demonstration	Install a C compiler.
Want to use Oracle Visual Information Retrieval with Oracle *inter*Media	Install Oracle *inter*Media first. If you used the Custom Installation type and selected both components for installation at the same time, Oracle *inter*Media was installed first.

Configuring Oracle *inter*Media Audio, Video, Image, and Locator

To configure Oracle *inter*Media Audio, Video, Image, and Locator:

1. Start SQL*Plus:

    ```
    C:\> SQLPLUS
    ```

2. Connect to the database with the SYS account:

    ```
    SQL> CONNECT SYS/PASSWORD AS SYSDBA
    ```

3. Start the database (if necessary):

    ```
    SQL> STARTUP
    ```

4. Run the ORDINST.SQL script:

    ```
    SQL> @ORACLE_BASE\ORACLE_HOME\ORD\ADMIN\ORDINST.SQL
    ```

5. Run the IMINST.SQL script:

    ```
    SQL> @ORACLE_BASE\ORACLE_HOME\ORD\IM\ADMIN\IMINST.SQL
    ```

6. Exit SQL*Plus:

    ```
    SQL> EXIT
    ```

Configuring Oracle *inter*Media, Audio, Video, Image, and Locator Demos

To configure Oracle *inter*Media Audio demonstrations:

1. Go to the *ORACLE_BASE**ORACLE_HOME*\ORD\AUD\DEMO directory.

2. Follow the instructions in the README.TXT file.

To configure Oracle *inter*Media Video demonstrations:

1. Go to the *ORACLE_BASE**ORACLE_HOME*\ORD\VID\DEMO directory.

2. Follow the instructions in the README.TXT file.

To configure Oracle *inter*Media Image demonstrations:

1. Go to the *ORACLE_BASE**ORACLE_HOME*\ORD\IMG\ADMIN directory.

2. Read the README file in this directory for instructions on configuring the demo.

3. Make the *inter*Media Image demos for a Microsoft C compiler by entering:

    ```
    C:\> MAKE
    ```

An additional demonstration resides in the following location:

```
ORACLE_BASE\ORACLE_HOME\ORD\IMG\DEMO\VC\IMGSAMP\SIMPIMG
```

In order to build and run the demonstration, you must first modify the MAKEFILE to adapt it to your environment.

To configure Oracle *inter*Media Locator demonstrations:

1. Go to the *ORACLE_BASE**ORACLE_HOME*\MD\DEMO\GEOCODER directory.

2. The NH_CS.SQL file contains the sample data that can be loaded into Oracle8*i*. GEOHTTP.SQL and GEOLOCAT.SQL are examples that show the use of Locator functionality. GEOINDEX.SQL contains examples of data indices created by using the Locator.

Oracle *inter*Media Text Configuration Responsibilities

Review the following table to determine your Oracle *inter*Media Text post-installation tasks.

If You...	Then...
Installed Oracle *inter*Media Text from the CD-ROM and you do not have a previous release of interMedia Text installed (formerly called ConText)	See the description below this table.
Installed Oracle *inter*Media Text from the CD-ROM and you do have a previous release of interMedia Text installed (formerly called ConText)	See *Oracle8i interMedia Text Migration.*
Migrated your database, you may need to configure Net8 for external procedures. Otherwise, Oracle *inter*Media Text may not work. In any case other than migration, Net8 should be configured correctly by default to work with Oracle *inter*Media Text.	See the *Oracle8i interMedia Text Migration* and *Oracle8i interMedia Text Reference.*
Are indexing formatted documents such as Microsoft Word	You must set your environment to use the INSO filter before you can index your documentation set. For more information on setting up your environment for INSO filtering, see Appendix C of the *Oracle8i interMedia Text Reference.*

If one of the following is true, the Oracle8*i* database is already configured for use with Oracle *inter*Media Text:

- The database is a starter database that you created by installing Oracle8*i* Enterprise Edition with the Typical installation type.

- The database is a starter database that you created by performing the following sequence of steps:

  1. Installed Oracle8*i* Enterprise Edition or Oracle8*i* with the Custom installation type.

  2. Selected Oracle8*i* Server in the *Available Product Components* dialog box.

  3. Clicked Yes when prompted to run Oracle Database Configuration Assistant.

  4. Selected the Typical database creation type.

  5. Selected Copy existing database files from the CD.

     Additional Information: See Chapter 6, "Post-Installation Database Creation" and Chapter 4 of *Oracle8i Installation Guide for Windows NT* for more information about creating a starter database.

- You created the database by using Oracle Database Configuration Assistant in standalone mode, selecting the Typical database creation type, and selecting Copy existing database files from the CD.

If none of these are true, you must configure the Oracle database for use with Oracle *inter*Media Text by doing one of the following:

- Using Oracle Database Configuration Assistant
- Configuring Manually

Using Oracle Database Configuration Assistant

You can use Oracle Database Configuration Assistant to configure an Oracle8*i* database for use with Oracle *inter*Media Text either at the time you create the database or later.

To configure...	Do this...
At a later time	1. Select Modify a database in the *Welcome* page.
	2. Select the database that you want to modify in the *Modify Instance* page.
	3. Select Oracle *inter*Media Text in the *Modify Database Cartridges and Options* page.

Configuring Manually

Manually configuring an Oracle database for use with Oracle *inter*Media Text consists of creating a tablespace for the Oracle *inter*Media Text data dictionary tables and then creating the CTXSYS user name and the Oracle *inter*Media Text data dictionary tables themselves.

To create a tablespace for the Oracle *inter*Media Text data dictionary tables:

1. Start SQL*Plus:

```
C:\> SQLPLUS
```

2. Connect as SYS:

```
Enter user-name: SYS/PASSWORD
```

3. Create a tablespace for the Oracle *inter*Media Text data dictionary tables:

```
SQL> CREATE TABLESPACE TABLESPACE_NAME DATAFILE 'ORACLE_BASE\ORADATA\
DB_NAME\DR01.DBF' SIZE 80M;
```

To create the CTXSYS user name and the *inter*Media Text data dictionary tables:

1. Connect as INTERNAL:

```
SQL> CONNECT INTERNAL/PASSWORD
```

2. Run the DR0CSYS.SQL script to create the CTXSYS user name:

```
SQL> @ORACLE_BASE\ORACLE_HOME\CTX\ADMIN\DR0CSYS.SQL PASSWORD
DEFAULT_TABLESPACE_NAME TEMPORARY_TABLESPACE_NAME;
```

where:

- *PASSWORD* is the password that you want to use for the CTXSYS user name.

- *DEFAULT_TABLESPACE_NAME* is the default tablespace for the Oracle *inter*Media Text data dictionary tables. Set the default tablespace to the value of *TABLESPACE_NAME* in step 3 of the instructions "To create a tablespace for the Oracle interMedia Text data dictionary tables:".

- *TEMPORARY_TABLESPACE_NAME* is the temporary tablespace for the Oracle *inter*Media Text data dictionary tables. Set the temporary tablespace to the value of *TABLESPACE_NAME* in step 3 of the instructions "To create a tablespace for the Oracle interMedia Text data dictionary tables:".

3. Connect as CTXSYS:

```
SQL> CONNECT CTXSYS/PASSWORD
```

4. Run the DR0INST.SQL script to create and populate the Oracle *inter*Media Text data dictionary tables:

```
SQL> @ORACLE_BASE\ORACLE_HOME\CTX\ADMIN\DR0INST.SQL ORACLE_BASE\
ORACLE_HOME\CTX\LIB\ORACTXX8.DLL;
```

5. Run the language-specific default script, where *XX* is the language code (for example, US):

```
SQL> @ORACLE_BASE\ORACLE_HOME\CTX\ADMIN\DEFAULTS\DRDEFXX.SQL;
```

6. Exit SQL*Plus:

```
SQL> EXIT
```

Oracle Spatial

Oracle Spatial makes the storage, retrieval, and manipulation of spatial data easier and more intuitive to users.

One example of spatial data is a road map. A road map is a two-dimensional object that contains points, lines, and polygons representing cities, roads, and political boundaries such as states. A road map represents geographic information. The locations of cities, roads, and political boundaries are projected onto a two-dimensional display or piece of paper, preserving the relative positions and relative distances of the objects.

Review the following table to determine your configuration responsibilities:

If You...	Then...
Installed Oracle Spatial through the Oracle8*i* Enterprise Edition Typical installation type	No manual configuration is required. All Oracle Spatial configuration tasks described under "Configuring Oracle Spatial" are automatically performed.
Installed both Oracle Spatial and Oracle8*i* Server together through the Oracle8*i* Enterprise Edition Custom installation type	Oracle Database Configuration Assistant starts at the end of installation. If you select either of the following options: ■ Custom ■ Typical, and then select the Create new database files suboption Oracle Database Configuration Assistant prompts you about whether or not you want Oracle Spatial to be automatically configured.
Installed Oracle Spatial during a separate installation from Oracle8*i* Enterprise Edition	You must manually configure Oracle Spatialby either: ■ Performing the Oracle Spatial configuration tasks described under "Configuring Oracle Spatial" ■ Starting Oracle Database Configuration Assistant and selecting Modify a database. Then following the on-screen instructions

Configuring Oracle Spatial

1. Start SQL*Plus at the MS-DOS command prompt:

    `C:\> SQLPLUS`

2. Connect to the database with the INTERNAL account:

    `Enter user-name: INTERNAL`

3. Start the database (if necessary):

    `SQL> STARTUP`

4. Run the ORDINST.SQL script:

    `SQL> @ORACLE_BASE\ORACLE_HOME\ORD\ADMIN\ORDINST.SQL`

5. Connect to the database as the SYSTEM user:

    `SQL> CONNECT SYSTEM/PASSWORD`

    where *PASSWORD* is MANAGER for the SYSTEM user account by default. If you have changed this password, substitute MANAGER with the correct password.

6. Run the MDINST.SQL script:

   `SQL> @ORACLE_BASE\ORACLE_HOME\MD\ADMIN\MDINST.SQL`

7. Exit SQL*Plus:

   `SQL> EXIT`

> **Note:** The script MDINST.SQL has a variable *%MD_SYS_ PASSWORD%* that is instantiated at installation time by Oracle Universal Installer. Therefore, if you have changed the MDSYS user's password, be sure during a manual installation to remember also to update the MDINST.SQL script with that password.

Oracle Time Series

Oracle Time Series stores and retrieves time-stamped data through object data types (ODTs).

Oracle Time Series is a building block for applications, rather than being an end-user application. For example, applications can use this option to process historical data derived from financial market transactions, such as trades of stocks, bonds, and mutual fund shares. From this you can find the opening, closing, low, and high prices for a stock on a specific date; calculate monthly volumes for a stock for a specific year; and derive the 30-day moving average for a stock over a year.

Oracle Time Series also comes as a set of demos that provide a sample demonstration of how the product works.

Review the following table to determine your configuration responsibilities.

If You...	Then...
Installed Oracle Time Series through the Oracle8*i* Enterprise Edition Typical installation type path	No manual configuration is required. All configuration tasks described under "Configuring Oracle Time Series" on page 5-11 are automatically performed.
Installed both Oracle Time Series and Oracle8*i* Server together through the Oracle8*i* Enterprise Edition Custom installation type path	Oracle Database Configuration Assistant starts at the end of installation. If you select either of the following options: ■ Custom ■ Typical, and then select the Create new database files suboption Oracle Database Configuration Assistant prompts you about whether or not you want Oracle Time Series to be automatically configured.
Installed Oracle Time Series during a *separate* installation from Oracle8*i* Enterprise Edition	You must manually configure Oracle Time Series by either: ■ Performing the tasks under "Configuring Oracle Time Series". ■ Starting Oracle Database Configuration Assistant and selecting Modify a database. Then following the onscreen instructions.
Want to use the demos	Perform the tasks described in the following table.
Manually copy your Oracle7 LISTENER.ORA and TNSNAMES.ORA files into your Oracle8*i* network directory	You must modify the TNSNAMES.ORA and LISTENER.ORA network configuration files on your server to enable external procedure calls to work and Oracle Time Series to function properly. Follow the tasks in the *Net8 Administrator's Guide*.

Configuring Oracle Time Series

To configure Oracle Time Series:

1. Start SQL*Plus:

    `C:\> SQLPLUS`

2. Connect to the database with the INTERNAL account:

    `Enter user-name: INTERNAL`

3. Start the database (if necessary):

    `SQL> STARTUP`

4. Run the ORDINST.SQL script:

    `SQL> @ORACLE_BASE\ORACLE_HOME\ORD\ADMIN\ORDINST.SQL`

5. Run the TSINST.SQL script:

```
SQL> @ORACLE_BASE\ORACLE_HOME\ORD\TS\ADMIN\TSINST.SQL
```

6. Exit SQL*Plus:

```
SQL> EXIT
```

The following Oracle Time Series demos are provided in subdirectories of *ORACLE_BASE\ORACLE_HOME*\ORD\TS\DEMO.

Configuring Oracle Time Series Demos

Demonstration	Directory	What the Demonstration Does
Basic usage	USAGE	Creates a sample database for use with Oracle Time Series and demonstrates several basic queries. This demo is used as the basis for all demos.
Option extension	EXTEND	Includes sample PL/SQL code to extend the functionality of Oracle Time Series with new functions. Oracle Objects option is required to extend Oracle Time Series.
OCI	OCI	Provides C examples of client-side time series access.
Pro*C/C++ Precompiler	PROC	Provides examples of several approaches to client-side time series access. Users must have Pro*C/C++ to build this demo.
Developer 2000	DEV2K	Includes a Developer 2000 Form that incorporates a Developer 2000 Graphic, both of which access data using Oracle Time Series. This demo requires Developer 2000 release 2.0 or later.
Quick Start	TSQUICK	Provides a quick start. This demo uses TSTools to automate the generation of the Oracle Time Series schema for a stock pricing database, then demonstrates several basic queries.
Specialized usage	USAGEUTL	Targets electrical utility applications. Demonstrates how to compute peak and off-peak summaries of 15-minute data.

To configure Oracle Time Series demos:

1. Go to the *ORACLE_BASE\ORACLE_HOME*\ORD\TS\DEMO directory.

2. Read the README file in this directory for an overview of the Oracle Time Series demos.

3. Go to the appropriate subdirectory of *ORACLE_BASE\ORACLE_HOME*\ORD\TS\DEMO for instructions on configuring the demo that you want to use (for example, directory PROC for Pro*C/C++).

4. Follow the README file instructions in the subdirectory to configure the demo.

> **Note:** Each of the subdirectories contains a README that provides configuration instructions.

Oracle Visual Information Retrieval

Oracle Visual Information Retrieval stores, retrieves, and manipulates image data managed by an Oracle8*i* database.

This option provides image storage, content-based retrieval, and format conversion capabilities through an object data type. This option is a building block for various imaging applications, rather than being an end-user application. Some common applications for this option consist of digital art galleries and museums, real estate marketing, document imaging, and stock photo collections for fashion designers and architects.

Oracle Visual Information Retrieval also provides a sample demonstration, showing how an image is extracted from an Oracle database.

Review the following table to determine your configuration responsibilities.

> **Note:** When you select installation of Oracle Visual Information Retrieval, Oracle *inter*Media is installed automatically since Oracle Visual Information Retrieval cannot function properly without it.

If You...	Then...
Installed Oracle Visual Information Retrieval through the Oracle8*i* Enterprise Edition Typical installation type	No manual configuration is required. All configuration tasks described under "To configure Oracle Visual Information Retrieval:" are automatically performed.
Installed both Oracle Visual Information Retrieval and Oracle8*i* Server together through the Oracle8*i* Enterprise Edition Custom installation type path	Oracle Database Configuration Assistant starts at the end of installation. If you select either of the following options: ■ Custom ■ Typical, and then select the Create new database files suboption Oracle Database Configuration Assistant asks if you want Oracle Visual Information Retrieval to be automatically configured.

If You...	Then...
Installed Oracle Visual Information Retrieval during a separate installation from Oracle8*i* Enterprise Edition	You must manually configure Oracle Visual Information Retrieval by either: ■ Performing the configuration tasks described under "Configuring Oracle Visual Information Retrieval" on page 5-14. ■ Starting Oracle Database Configuration Assistant and selecting "Modify a database". Then, following the on-screen instructions.
Want to use the demonstration	Perform the configuration tasks described under "Configuring Oracle Visual Information Retrieval".
Want to build an Oracle Visual Information Retrieval sample demonstration.	Install a C compiler.
Manually copy your Oracle7 LISTENER.ORA and TNSNAMES.ORA files into your Oracle8i network directory	You must modify the TNSNAMES.ORA and LISTENER.ORA network configuration files on your server to enable external procedure calls to work and Oracle Visual Information Retrieval to function properly. Follow the tasks in the *Net8 Administrator's Guide*.

Configuring Oracle Visual Information Retrieval

To configure Oracle Visual Information Retrieval:

1. Ensure that Oracle *inter*Media is already configured. Oracle *inter*Media must be configured before Oracle Visual Information Retrieval. See the configuration instructions for Oracle interMedia under "Configuring Oracle interMedia, Audio, Video, Image, and Locator Demos" on page 5-4 for information about running the ordinst.sql and iminst.sql scripts.

2. Start SQL*Plus:

   ```
   C:\> SQLPLUS
   ```

3. Connect to the database with the SYS account:

   ```
   SQL> CONNECT SYS/PASSWORD AS SYSDBA
   ```

4. Start the database (if necessary):

   ```
   SQL> STARTUP
   ```

5. Run the VIRINST.SQL script:

   ```
   SQL> @ORACLE_BASE\ORACLE_HOME\ORD\VIR\ADMIN\VIRINST.SQL
   ```

6. Run the IMINST.SQL script:

   ```
   SQL> @ORACLE_BASE\ORACLE_HOME\ORD\IM\ADMIN\IMINST.SQL
   ```

7. Exit SQL*Plus:

```
SQL> EXIT
```

To configure the Oracle Visual Information Retrieval demonstration:

1. Go to the *ORACLE_BASE\ORACLE_HOME*\ORD\VIR\ADMIN directory.

2. Read the README file in this directory for instructions on configuring the demonstration.

3. Make the Oracle Visual Information Retrieval demonstration for a Microsoft C compiler by entering:

```
C:\> MAKE
```

Multithreaded Server Support

Oracle Database Configuration Assistant lets you enable or disable multithreaded server support in your Oracle8*i* database.

Multithreaded server mode is also called shared server mode.

If your Oracle8*i* database is not configured for multithreaded server mode, then it is configured for dedicated server mode.

The following table describes the differences between dedicated server mode and multithreaded server mode.

Mode	Description
Dedicated server mode	The Oracle8*i* database allocates a resource dedicated to serving only that one client connection. This mode is best used in the following environments: ■ Warehousing environment. ■ A small number of users will connect to your Oracle8*i* database.

Mode	Description
Multithreaded server mode (also called shared server mode)	Enables many client user processes to share a small number of server processes.
	Many client users can connect to a dispatcher process. The dispatcher process then routes client requests to the next available shared server process. No dedicated server process exists for each client user process for the duration of the connection. Instead, inactive server processes are recycled and used as needed. This reduces system overhead and enables you to increase the number of supported users.
	This mode is best used in the following environments:
	■ Online transaction processing (OLTP) environment
	■ A large number of users will simultaneously connect to your database
	■ You want to use Net8 features such as connection pooling, connection multiplexing, and load balancing
	■ Managing and using system resources to a high degree is important
	■ Predictable and fast database connection times are very important. This may be a very important criterion for Web applications.
	Note: This mode is required for Oracle JServer to function properly.

> **See Also:** *Net8 Administrator's Guide* for more information about the multithreaded server mode.

There are two types of multithreaded server support:

■ Internet Inter-ORB Protocol (IIOP) clients

■ Two-task Net8 clients

These types are independent of each other. In other words, you can have any of the following combinations:

■ Both types of support are enabled

■ Both types of support are disabled

■ One type of support is enabled, and the other type of support is disabled

The current configuration of your Oracle8*i* database depends on how the database was installed.

If installed through...	Then the configuration is...
Oracle8*i* Enterprise Edition or Oracle8*i* Typical installation type of Oracle Universal Installer	multithreaded server mode for IIOP clients and dedicated server mode for two-task Net8 clients
Oracle8*i* Enterprise Edition Minimal installation type of Oracle Universal Installer	Dedicated server mode for both types of clients
Typical option of Oracle Database Configuration Assistant	If you selected Oracle JServer, the mode is multithreaded server mode for IIOP clients.
	Dedicated server mode for two-task Net8 clients, unless you perform the following sequence of steps:
	1. Run Oracle Database Configuration Assistant.
	2. Select Create a database.
	3. Select Typical.
	4. Select the Create new database files suboption.
	5. Select Online Transaction Processing (OLTP) as your database environment.
	6. Enter 20 or more for the number of concurrent database connections.
	This creates a database in multithreaded server mode for two-task Net8 clients.
Custom option of Oracle Database Configuration Assistant	Dedicated server mode or multithreaded server mode, depending on what you select when prompted by Oracle Database Configuration Assistant.

Enabling Multithreaded Server Support for IIOP Clients

Multithreaded server support for IIOP clients is automatically enabled when you install Oracle JServer.

Enabling Multithreaded Server Support for Two-Task Net8 Clients

Multithreaded support for two-task Net8 clients must be enabled manually.

To enable multithreaded server support for two-task Net8 clients:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Database Configuration Assistant.

The *Oracle Database Configuration Assistant Welcome* page appears.

2. Select Modify a database and click Next.

3. Select the Oracle8*i* database to modify and, if prompted, enter the INTERNAL password.

4. Click Next.

5. Select Shared Server Mode and click Next.

6. Click Next in the *Oracle Database Configuration Assistant Welcome* page.

7. Make any necessary changes to the multithreaded server parameters. Click Help for more information about the parameters.

8. Click Next.

9. Make any necessary changes to additional multithreaded server parameters. Click Help for more information about the parameters.

10. Click Finish.

    A dialog box prompts you to select the initialization parameter file to use.

11. Select the appropriate file and click OK.

    Your initialization parameter file is modified.

12. Shut down and restart your Oracle8*i* database for the changes to take effect.

Disabling Multithreaded Server Support for Two-Task Net8 Clients

Multithreaded support for two-task Net8 clients must be disabled manually.

To disable multithreaded server support for two-task Net8 clients:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Database Configuration Assistant.

    The *Oracle Database Configuration Assistant Welcome* page appears.

2. Select Modify a database and click Next.

3. Select the Oracle8*i* database to modify and, if prompted, enter the INTERNAL password.

4. Click Next.

5. Select Dedicated Server Mode and click Finish.

    A dialog box prompts you to select the initialization parameter file to use.

6. Select the appropriate file and click OK.

   Your initialization parameter file is modified.

7. Shut down and restart your Oracle8*i* database for the changes to take effect.

Advanced Replication

This section describes how to configure Advanced Replication in your Oracle8*i* database.

The following figure describes the situations under which Advanced Replication is configured or not configured.

Follow the instructions *only* if you want to add Advanced Replication to an Oracle8*i* database that was not previously configured with this feature.

If you select the Oracle8*i* Enterprise Edition top-level component and then select...

Typical	Minimal	Custom

Advanced replication is automatically configured.

Advanced replication is automatically configured *if* you installed the starter database when prompted by the *Select Starter Database* window.

The *Available Product Components* window appears.

1. Select Oracle Server and click Install.
2. Select Yes when prompted to run Oracle Database Configuration Assistant.

Oracle Database Configuration Assistant starts at the end of installation and prompts you to select either of two database installation types:
· Typical
· Custom

If you select Typical, Oracle Universal Installer prompts you about how to create your database:
· Copy existing database files from the CD
· Create new database files

If you select Custom, one of the Oracle Database Configuration Assistant pages prompts you to configure Advanced Replication.

If you select the Advanced Replication check box, advanced replication is automatically configured.

If you select "Copy existing database files from the CD", advanced replication is automatically configured.

If you select "Create new database files", one of the Oracle Database Configuration Assistant pages prompts you to configure Advanced Replication.

If you select the Advanced Replication check box, advanced replication is automatically configured.

Configuring Advanced Replication consists of the following steps:

- Step 1: Checking Tablespace Requirements

- Step 2: Checking Initialization Parameters

- Step 3: Performing Configuration Tasks

- Step 4: Monitoring Data Dictionary Tables

- Step 5: Upgrading Advanced Replication

> **See Also:** There are many configuration and usage possibilities with Advanced Replication. For more information about Advanced Replication and for definitions of master sites and snapshot sites, see the following guides:
>
> - *Oracle8i Distributed Database Systems*
>
> - *Oracle8i Replication*
>
> - *Oracle8i Concepts*
>
> - *Oracle8i Administrator's Guide*

Step 1: Checking Tablespace Requirements

The following are recommended tablespace requirements for Advanced Replication:

Tablespace	Requirement
SYSTEM	At least 20 MB of free space is required for replication packages. Replication triggers and procedures are stored here.
ROLLBACK SEGMENTS	■ Rollback Initial Extent = 50 KB ■ Rollback Next Extent = 50 KB
ROLLBACK	At least 5 MB of free space.
TEMPORARY	At least 10 MB of free space.
USER	No specific requirement.

Step 2: Checking Initialization Parameters

Certain initialization parameter values must be set or added to the INIT.ORA file (recommended values are included) if you use Advanced Replication.

Master Site

Add these initialization parameters to the master site.

Parameter Name	Recommended Value
JAVA_POOL_SIZE	20 MB
DISTRIBUTED_LOCK_TIMEOUT	300 seconds
DISTRIBUTED_TRANSACTIONS	5
GLOBAL_NAMES	TRUE
OPEN_LINKS	4
PROCESSES	Add 9 to current value
JOB_QUEUE_PROCESSES	2 (depends on number of n-way sites)
JOB_QUEUE_INTERVAL	10 s

Snapshot Sites

Add these initialization parameters for snapshot sites.

Parameter Name	Recommended Value
JOB_QUEUE_PROCESSES	2
JOB_QUEUE_INTERVAL	60 s

Step 3: Performing Configuration Tasks

After setting up the INIT.ORA file with the initialization parameters for Advanced Replication, perform the following steps:

To configure Advanced Replication:

1. Start SQL*Plus:

    ```
    C:\> SQLPLUS
    ```

2. Connect to the database with the INTERNAL account:

    ```
    Enter user-name: INTERNAL
    ```

3. Start the database if it is not currently running:

    ```
    SQL> STARTUP
    ```

4. Spool output to a log file first before running the necessary SQL script.

    Enter the following command:

    ```
    SQL> SPOOL OUTPUT.LOG
    ```

    This creates a file called OUTPUT.LOG to which to output all on-screen activity while the SPOOL session is open.

5. Run the CATREP.SQL script once the database has started. The script is found in the *ORACLE_BASE\ORACLE_HOME*\RDBMS\ADMIN directory, and takes approximately one hour to run.

    ```
    SQL> @ORACLE_BASE\ORACLE_HOME\RDBMS\ADMIN\CATREP.SQL
    ```

6. Close the SPOOL file when the script has run completely:

    ```
    SQL> SPOOL OFF
    ```

    OUTPUT.LOG is saved to your current directory.

7. Confirm that CATREP.SQL ran correctly by running a query on ALL_OBJECTS where STATUS = 'INVALID':

    ```
    SQL> SELECT * FROM ALL_OBJECTS WHERE STATUS = 'INVALID';
    ```

    If all package bodies compiled successfully, the following message displays:

    ```
    0 rows selected.
    ```

    If you find that any of the package bodies compiled incorrectly, recompile them manually. The syntax for running them manually follows:

    ```
    SQL> ALTER PACKAGE PACKAGE_NAME COMPILE BODY;
    ```

    If CATREP.SQL ran successfully, a number of replication tables are created in the SYSTEM tablespace. The database is now set up for Advanced Replication support.

8. Exit SQL*Plus:

   ```
   SQL> EXIT
   ```

9. See *Oracle8i Replication* for details on setting up the following:

   - Master definition site

   - Master sites

   - Updatable snapshot sites

   - Conflict resolution

   > **Note:** Run the following statement on the master site when setting up updatable snapshot sites:
   >
   > ```
   > SVRMGR> GRANT EXECUTE ON DBMSOBJGWRAPPER TO PUBLIC;
   > ```

Step 4: Monitoring Data Dictionary Tables

The practical limit to the number of master sites an environment can have is 36. The processes SNP0 to SNP9 (total of 10) and SNPA to SNPZ (total of 26) each handle one destination master at a time.

If you use Advanced Replication and intend to set up a large number of replicated objects, monitor the following data dictionary tables with the SQL SELECT command:

- ARGUMENT$

- IDL_CHAR$

- IDL_UB1$

- IDL_UB2$

- IDL_SB4$

- I_ARGUMENT1

- I_SOURCE1I$

- SOURCE$

- TRIGGER

If necessary, increase the storage parameters to accommodate the storage requirements of large numbers of replicated objects.

Step 5: Upgrading Advanced Replication

If you are upgrading from a previous version of the Oracle database and want to use Advanced Replication, you must first complete the upgrade. See *Oracle8i Migration.*

Because advanced replication is an advanced feature, see *Oracle8i Replication* for a presentation of its basic concepts before proceeding. For information about available training, contact Oracle Education or see Oracle *Metalink.*

6

Post-Installation Database Creation

This chapter describes how to create a database with Oracle Database Configuration Assistant or the BUILD_DB.SQL script after installing Oracle.

Specific topics discussed are:

- Before You Create a Database

- Creating a Database Using Tools

- Using Oracle Database Configuration Assistant

- Using BUILD_DB.SQL

- Using ORADIM to Administer an Oracle InstanceUsing ORADIM to Administer an Oracle Instance

Before You Create a Database

Before you create a database, consider the following requirements described below.

Naming Conventions for Oracle Databases

With Oracle8*i* database, all mounted Oracle databases in a network *must* have unique database names.

A name is associated with a database at database creation time and stored in its control files. If the database keyword is provided in the CREATE DATABASE statement or when prompted by the Oracle Database Configuration Assistant, that value becomes the name for that database. If not, the program uses the value of the DB_NAME parameter in the INIT.ORA file.

If you attempt to mount two Oracle8*i* databases with the same database name, you receive the following error during the second mount:

```
ORA-01102: cannot mount database in EXCLUSIVE mode
```

If there are two or more Oracle8*i* databases on the same computer, but located in different Oracle homes, the following rules apply:

■   Each database name must be unique

■   Each SID must be unique

To change the name of an existing database, you must use the CREATE CONTROLFILE statement to recreate your control file(s) and specify a new database name. This restriction only exists for Oracle8*i* instances. Any Oracle7 instances running simultaneously with an Oracle8*i* instance are not subject to this restriction.

Note: The directory path examples in this chapter follow Optimal Flexible Architecture (OFA) guidelines (for example, *ORACLE_ BASE\ORACLE_HOME*\RDBMS\ADMIN). If you specified non-OFA compliant directories during installation, your directory paths will differ. See "OFA and Multiple Oracle Home Configurations" on page 3-23 for information.

Creating Data Files and Log Files on Remote Computers

Although it is possible for Oracle to access database files on remote computers using Universal Naming Convention (UNC), it is not recommended because of performance and network reliability concerns.

UNC is a PC format for specifying the location of resources on a local area network. UNC uses the following format:

```
\\SERVER-NAME\SHARED-RESOURCE-PATHNAME
```

For example, to access the file SYSTEM01.DBF in the directory C:\ORACLE\ORADATA\ORCL on the shared server ARGON, you reference the file as:

```
\\ARGON\ORACLE\ORADATA\ORCL\SYSTEM01.DBF
```

Note that the location of archive log files cannot be specified using UNC. If you set the LOG_ARCHIVE_DEST_n initialization parameter to a UNC specification, the database does not start and you receive the following errors:

```
ORA-00256: error occurred in translating archive text string '\meldell\rmdrive'
```

```
ORA-09291: sksachk: invalid device specified for archive destination
```

```
OSD-04018: Unable to access the specified directory or device
```

```
O/S-Error: (OS 2) The system cannot find the file specified
```

Ensure that you set the LOG_ARCHIVE_DEST_n initialization parameter to a mapped drive.

Note: An ORA-00256 error also occurs if you enter:

```
\\\meldell\rmdrive
```

or

```
\\\meldell\\rmdrive
```

Control files required the additional backslashes for release 8.0.4, but redo log files and data files did not.

Creating a Database Using Tools

You can choose either of the following tools to create a database:

- Oracle Database Configuration Assistant
- BUILD_DB.SQL script

Use Oracle Database Configuration Assistant to create a database, because it is the easier method.

If you want to create a database using command line tools, you can use the BUILD_DB.SQL script located in *ORACLE_BASE**ORACLE_HOME*\RDBMS\ADMIN.

Using Oracle Database Configuration Assistant

Oracle Database Configuration Assistant enables you to:

- Create a Database
- Change Database Configuration
- Delete a Database

> **Note:** This chapter describes running Oracle Database Configuration Assistant in standalone mode (that is, after installation). See Chapter 4 of the *Oracle8i Installation Guide for Windows NT* for information on running Oracle Database Configuration Assistant during installation to create a database.

Create a Database

> **Note:** If you use Oracle Database Configuration Assistant to create a new database in a new multiple Oracle home, the LISTENER.ORA file located in *ORACLE_HOME*\NETWORK\ADMIN is updated with the SID information. Also a new TNS entry is generated in the TNSNAMES.ORA file located in *ORACLE_HOME*\NETWORK\ADMIN.

To create a database using Oracle Database Configuration Assistant:

> **Note:**   Users must have Windows NT Administrator's privileges
> in order to create an Oracle8*i* database. If Oracle Database
> Configuration Assistant is run from an account that is not part of
> the Administrator group it gives an appropriate warning stating
> that you don't have administrative privileges to create the database.
> Log in as a user that is part of the Administrator groupand restart
> this tool again to create the database.

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration
> Database Configuration Assistant.

The Oracle Database Configuration Assistant Welcome page appears.

2. Select Create a database, then click Next.

The following page appears:

Option	Description
Typical	Consists of two suboptions:
	■ Copy existing database files from the CD
	Automatically installs a Hybrid starter database with the default initialization parameter settings
	■ Create new database files
	Asks you several database environment questions before dynamically creating a database

Option	Description
Custom	Enables you to customize the creation of your database. This option is only for Oracle database administrators (DBAs) experienced with advanced database creation procedures, such as customizing: ■ Data, control, and redo log file settings ■ Tablespace sizes ■ Extent sizes ■ Database memory parameters ■ Archiving formats and destinations ■ Trace file destinations ■ Character set values

3. Choose the Typical or Custom option to create a database:

Both the Typical option (through its Create new database files suboption) and the Custom option enable you to specify the type of environment in which to operate your Oracle8*i* database:

Environment	Description
Online Transaction Processing (OLTP)	Many concurrent users performing numerous transactions requiring rapid access to data. Availability, speed, concurrence, and recoverability are key issues.
	Transactions consist of reading (SELECT statements), writing (INSERT and UPDATE statements), and deleting (DELETE statements) data in database tables.
Warehousing	Users perform numerous, complex queries that process large volumes of data. Response time, accuracy, and availability are key issues.
	These queries (typically read-only) range from a simple fetch of a few records to numerous complex queries that sort thousands of records from many different tables. Warehousing environments are also known as Decision Support System (DSS) environments.
Multipurpose	Both types of applications(OLTP or warehousing) can access this database.

4. Respond to instructions on each Oracle Database Configuration Assistant page, then click Next when you are ready to continue to the next page. When you get to the last page, click Finish to start the creation of the Oracle8*i* database.

Importing Sample Schemas

Sample OLTP and Warehousing database schemas are available on the CD-ROM. After you finish creating your Oracle8*i* database with Oracle Database Configuration Assistant, you can import the appropriate sample schema.

If you selected multipurpose, you do not need to import a sample schema; one is already provided with your database.

To import an OLTP or DSS sample schema into your Oracle8*i* database:

1. Start SQL*Plus:

   ```
   C:\> SQLPLUS
   ```

2. Connect with the SYSTEM account:

   ```
   Enter user-name: SYSTEM/PASSWORD
   ```

3. Create a special user account for importing the appropriate schema:

   ```
   SQL> CREATE USER SAMPLE_USER IDENTIFIED BY PASSWORD;

   SQL> GRANT RESOURCE TO SAMPLE_USER;

   SQL> GRANT CONNECT TO SAMPLE_USER;
   ```

   where *SAMPLE_USER* is SAMPLEOLTP for the OLTP sample schema or SAMPLESTAR for the DSS sample schema.

4. Exit SQL*Plus:

   ```
   SQL> EXIT
   ```

5. Go to the *ORACLE_BASE\ORACLE_HOME*\ASSISTANTS\DBCA\SAMPLES directory on your hard drive.

6. Import the appropriate schema:

```
C:\ORACLE\ORA81\ASSISTANTS\DBCA\SAMPLES> IMP SAMPLE_USER/PASSWORD
FILE=SAMPLE.DMP FULL=Y LOG=MYIMP.LOG
```

where:

C:\ORACLE	is the *ORACLE_BASE* directory
ORA81	is the *ORACLE_HOME* directory
SAMPLE_USER	is SAMPLEOLTP for the OLTP sample schema or SAMPLESTAR for the Warehousing sample schema
SAMPLE.DMP	is SOURCE80.DMP for the OLTP sample schema or TARGET80.DMP for the Warehousing sample schema

Change Database Configuration

The Change Database Configuration option enables you to perform the following procedures:

- Configuring Advanced Replication and Oracle options
- Enabling and Disabling MultiThreaded Server Support

Configuring Advanced Replication and Oracle options

This enables an Oracle8*i* database to support Advanced Replication functionality and the following as-yet-unconfigured options that you installed from your CD-ROM:

- Oracle *inter*Media:
  - Oracle *inter*Media Audio
  - Oracle *inter*Media Video
  - Oracle *inter*Media Locator
  - Oracle *inter*Media Image
  - Oracle *inter*Media Text
- Oracle Spatial
- Oracle Time Series
- Oracle Visual Information Retrieval
- Oracle JServer

- SQL*Plus Help

- Advanced Replication

These options (if installed during a separate installation from Oracle8*i* Enterprise Edition) are not automatically configured during installation. If you installed Oracle options through the Oracle8*i* Enterprise Edition Typical installation type on the CD-ROM, your options were automatically configured for the starter database.

> **Note:** If you installed Oracle Visual Information Retrieval and its check box is greyed-out, select the check box for Oracle *inter*Media Image. Doing this makes the check box for Oracle Visual Information Retrieval selectable. This is because Oracle Visual Information Retrieval is dependent upon Oracle *inter*Media Image. In addition, JServer is a required dependency for Oracle *inter*Media Image and Oracle Visual Information Retrieval.

Enabling and Disabling MultiThreaded Server Support This lets you choose whether to enable or disable multithreaded server support in your Oracle8*i* database.

Multithreaded server (MTS) support enables multiple client user processes to share a small number of server processes. Many client users can connect to a dispatcher process. The dispatcher process then routes client requests to the next available shared server process. No dedicated server process exists for each remaining user process associated with the client user process for the duration of the connection. Instead, inactive server processes are "recycled" and used as needed. This reduces system overhead and enables you to increase the number of supported users.

> **Additional Information:** See "Multithreaded Server Support" on page 5-15.
>
> In addition, see the following guides:
>
> - *Net8 Administrator's Guide*
>
> - *Oracle8i Concepts*
>
> - *Oracle8i Reference*

Delete a Database

The Delete a Database option of Oracle Database Configuration Assistant lets you quickly and easily delete all database files including the initialization parameter file.

Using BUILD_DB.SQL

This section describes how to create a new database manually using a SQL script. There are a number of ways to create a database depending on if you want to:

- Make a copy of an existing database and remove the old database.

- Make a copy of an existing database and keep the old database.

- Create a new database when no database exists on your system that you can copy.

The following table summarizes the steps involved in creating a new database for each of these database creation scenarios. Each step is explained in detail in the following subsections.

Perform these tasks...	If you want to...		
	Copy an existing database to a new database, then remove the old database	Copy an existing database to a new database, then keep the old database	Create a new database when no other database exists on your system
Exporting an Existing Database	Yes	Only if you want to copy data from the existing database to the new database	Not applicable
Deleting Database Files	Yes	No	Not applicable
Modifying the INIT.ORA File	Yes	Yes	Yes
Creating and Starting an Oracle Service	No	Yes	Yes
Putting the CREATE DATABASE Statement in a Script	Yes	Yes	Yes
Creating a Database	Yes	Yes	Yes
Importing a Database	Yes	Only if you want to import tables and other objects exported from the existing database	Not applicable
Updating the ORACLE_SID in the Registry	No	Only if you want to change the default SID	Yes
Backing Up the New Database	Yes	Yes	Yes

How to Create a Database

An example is used in the following sections to demonstrate how to create a database.

In this example, you will copy an existing database (the starter database with a SID of ORCL located in the C:\ORACLE\ORADATA\ORCL directory) to a new

database with a database name and SID of PROD located in the C:\ORACLE\ORADATA\PROD directory.

You will delete the starter database ORCL after you have created the PROD database.

> **Note:** In this example, *ORACLE_BASE* is C:\ORACLE. See Chapter 3, "Multiple Oracle Homes and Optimal Flexible Architecture" for more information on *ORACLE_BASE*.

Creating Directories

Create the following directories in which to put the administration and database files for the new database PROD:

- C:\ORACLE\ADMIN\PROD

- C:\ORACLE\ADMIN\PROD\BDUMP

- C:\ORACLE\ADMIN\PROD\PFILE

- C:\ORACLE\ADMIN\PROD\UDUMP

- C:\ORACLE\ORADATA\PROD

Exporting an Existing Database

You only need to export an existing database if you want to copy its contents to a new database.

You can invoke the Export Utility by using either parameter mode or interactive mode. However, parameter mode is the recommended mode. Interactive mode provides less functionality than parameter mode and exists for backward compatibility only.

Example 6–1   Parameter Mode

```
C:\> EXP SYSTEM/PASSWORD FILE=MYEXP.DMP FULL=Y LOG=MYEXP.LOG
```

Example 6–2   Interactive Mode

```
C:\> EXP SYSTEM/PASSWORD
```

Enter only the command EXP SYSTEM/*PASSWORD* to begin an interactive session and let the Export Utility prompt you for the information it needs.

See *Oracle8i Utilities* for more information on using the Export Utility.

> **Note:** If you use parameter mode, the Export Utility considers file names and directory names to be invalid if a blank space is present. Enclose the full path in the FILE= parameter in triple quotes. For example:
>
> ```
> FILE="""C:\PROGRAM FILES\EXPORT.DMP"""
> ```
>
> or
>
> ```
> FILE="""C:\PROGRAM FILES\EXPORT FILE.DMP"""
> ```
>
> If the Export Utility is used in interactive mode, the file name or directory name can contain a space without quotes.

To export all data from an existing database to a new database:

1. Set ORACLE_SID to the database service of the database whose contents you want to export. For example, if the database you want to export is the starter database ORCL, enter the following at the MS-DOS command prompt. Note that there are no spaces around the equal sign (=) character.

   ```
   C:\> SET ORACLE_SID=ORCL
   ```

2. Start the Export Utility from the MS-DOS command prompt:

   ```
   C:\> EXP SYSTEM/PASSWORD FILE=MYEXP.DMP FULL=Y LOG=MYEXP.LOG
   ```

   You now have a full database export of the starter database ORCL in the file MYEXP.DMP. All messages from the Export Utility are logged in the file MYEXP.LOG.

Deleting Database Files

Deleting database files is only required when you want to copy an existing database to a new database to replace the old database. In the following example, you delete the database files of the starter database ORCL.

To delete database files:

1. Shut down the starter database ORCL at the MS-DOS command prompt:

   ```
   C:\> ORADIM –SHUTDOWN –SID ORCL –USRPWD PASSWORD –SHUTTYPE INST

   –SHUTMODE I
   ```

2. Delete the following database files located in the
C:\ORACLE\ORADATA\ORCL directory:

- CONTROL01.CTL

- CONTROL02.CTL

- CONTROL03.CTL

- INDX01.DBF

- DR01.DBF

- RBS01.DBF

- SYSTEM01.DBF

- TEMP01.DBF

- USERS01.DBF

- REDO01.LOG

- REDO02.LOG

- REDO03.LOG

- TOOLS01.DBF

Modifying the **INIT.ORA** File

If you are using the starter database ORCL as the basis for your new database, copy
INIT.ORA file:

C:\ORACLE\AORACLE_HOME\DMIN\ORCL\PFILE\INIT.ORA

to

C:\ORACLE\ORACLE_HOME\ADMIN\PROD\PFILE\INIT.ORA

and modify the file as described in this section.

If you do not have an existing database on your system, you cannot copy an
initialization parameter file to use as the basis for your new INIT.ORA file.
However, you can use the sample initialization parameter file INITSMPL.ORA
provided in the *ORACLE_BASE*\ORACLE_HOME\ADMIN\SAMPLE\PFILE
directory as the basis for the INIT.ORA file for the PROD database.

If you use INITSMPL.ORA as the basis for the INIT.ORA file, you must modify the following initialization parameters in the INIT.ORA file, or you will not be able to start the PROD database:

- DB_NAME

- INSTANCE_NAME

- SERVICE_NAMES

- CONTROL_FILES

- BACKGROUND_DUMP_DEST

- USER_DUMP_DEST

Modifying the DB_FILES initialization parameter is recommended to optimize performance.

Initialization Parameter	Modification Instructions.
DB_NAME	This parameter indicates the name of the database, and must match the name used in the CREATE DATABASE statement in "Putting the CREATE DATABASE Statement in a Script" on page 6-17. You give a unique database name to each database. You can use up to eight characters for a database name. The name does not need to match the SID of the database service.
	Set this parameter to:
	`DB_NAME=PROD.DOMAIN`
INSTANCE_NAME	Set this parameter to:
	`INSTANCE_NAME=PROD.DOMAIN`
SERVICE_NAMES	Set this parameter to:
	`SERVICE_NAMES=PROD.DOMAIN`

Initialization Parameter	Modification Instructions.
CONTROL_FILES	This parameter lists the control files of the database. You do not have the control files on your file system at this point, because the control files are created when you run the CREATE DATABASE statement. Ensure that you specify the complete path and file name, including drive letter.
	Set this parameter to:
	`CONTROL_FILES = ("C:\ORACLE\ORADATA\PROD\CONTROL01.CTL", "C:\ORACLE\ORADATA\PROD\CONTROL02.CTL", "C:\ORACLE\ORADATA\PROD\CONTROL03.CTL")`
BACKGROUND_DUMP_DEST	Set this parameter to:
	`BACKGROUND_DUMP_DEST = C:\ORACLE\ADMIN\PROD\BDUMP`
USER_DUMP_DEST	Set this parameter to:
	`USER_DUMP_DEST = C:\ORACLE\ADMIN\PROD\UDUMP`
DB_FILES	Set this parameter to the same number as the value of the MAXDATAFILES option of the CREATE DATABASE statement. The value of 100 is used for this example.
	`DB_FILES=100`

> **See Also:** Appendix B, "Oracle8i Database Specifications for Windows NT" and *Oracle8i Reference* for information on other initialization parameters that you may want to add or modify.

Creating and Starting an Oracle Service

You only need to create and start an Oracle service if you want to do one of the following:

- Copy an existing database to a new database and keep the old database

- Create a new database when no other database exists on your system that you can copy

Before you create the database, first create a Windows NT service to run the database. This service is the Oracle8i database process, ORACLE.EXE, installed in the form of a Windows NT service.

Use ORADIM to create the service. After it has been created, the service starts automatically. See "Using ORADIM to Administer an Oracle Instance" on page 6-26 for information on how to use ORADIM.

To create and start an Oracle service:

1. Run ORADIM from the MS-DOS command prompt:

   ```
   C:\> ORADIM -NEW -SID PROD -INTPWD PASSWORD -STARTMODE MANUAL

   -PFILE C:\ORACLE\ADMIN\PROD\PFILE\INIT.ORA
   ```

   Note that the previously-created INIT.ORA file is specified, with complete path, including drive name. You can check if the service is started in the services window of the Windows NT Control Panel.

2. Set ORACLE_SID to equal PROD. Note that there are no spaces around the equal sign (=) character:

   ```
   C:\> SET ORACLE_SID=PROD
   ```

Putting the CREATE DATABASE Statement in a Script

The CREATE DATABASE statement is a sequence of SQL statements that creates the database. Create a script containing this statement that you can reuse anytime you want to create a database.

Open the BUILD_DB.SQL script located in C:\ORACLE\ORA81\RDBMS\ADMIN and save it as BUILD_PROD.SQL.

This file becomes the basis for your script.

> **Note:**   The following example uses the BUILD_DB.SQL script to create a database. You can also use the BUILDALL.SQL script to create a database. BUILDALL.SQL not only creates the database by calling BUILD_DB.SQL but also runs many other scripts such as CATALOG.SQL, CATSNMP.SQL, SCOTT.SQL, and COMDEMO.SQL.

To prepare the CREATE DATABASE script:

Make the following changes to the BUILD_PROD.SQL script.

1. Set PFILE so it points to the C:\ORACLE\ADMIN\PROD\PFILE\INIT.ORA initialization file.

2. Change CREATE DATABASE SAMPLE to CREATE DATABASE PROD.

3. Change all occurrences of SAMPLE to PROD. For example, change
C:\ORACLE\ORADATA\SAMPLE\REDO01.LOG to
C:\ORACLE\ORADATA\PROD\REDO01.LOG

The following is the sample BUILD_DB.SQL script. Areas that you must modify to
create a database called PROD are highlighted.

```
--
-- This file must be run out of the directory containing the
-- initialization file.

startup nomount pfile=C:\Oracle\ADMIN\SAMPLE\pfile\initsmpl.ora

--  Create database

create database SAMPLE
    controlfile reuse
    logfile 'C:\Oracle\ORADATA\SAMPLE\redo01.log' size 1M reuse,
            'C:\Oracle\ORADATA\SAMPLE\redo02.log' size 1M reuse,
            'C:\Oracle\ORADATA\SAMPLE\redo03.log' size 1M reuse
                datafile 'C:\Oracle\ORADATA\SAMPLE\system01.dbf' size 10M reuse
autoextend on
next 10M maxsize 200M
character set WE8ISO8859P1;

create rollback segment rb_temp storage (initial 100 k next 250 k);

-- Create additional tablespaces ...

-- USERs: Create user sets this as the default tablespace
-- TEMP: Create user sets this as the temporary tablespace
-- RBS: For rollback segments

create tablespace users
    datafile 'C:\Oracle\ORADATA\SAMPLE\users01.dbf' size 3M reuse autoextend on
      next 5M maxsize 150M;
create tablespace rbs
    datafile 'C:\Oracle\ORADATA\SAMPLE\rbs01.dbf' size 5M reuse autoextend on
      next 5M maxsize 150M;
create tablespace temp
    datafile 'C:\Oracle\ORADATA\SAMPLE\temp01.dbf' size 2M reuse autoextend on
      next 5M maxsize 150M;
create tablespace oem_repository
    datafile 'C:\Oracle\ORADATA\SAMPLE\oemrep01.dbf' size 3M reuse autoextend on
      next 5M maxsize 150M;
create tablespace indx;
    datafile 'C:\Oracle\ORADATA\SAMPLE\indx01.dbf' size 2M reuse autoextend on
      next 5M maxsize 150M;
--      next 5M maxsize 150M;
```

```
alter rollback segment rb_temp online;

-- Change the SYSTEM users' password, default tablespace and
-- temporary tablespace.

alter user system temporary tablespace temp;
alter user system default tablespace users;

-- Create 16 rollback segments.  Allows 16 concurrent users with open
-- transactions updating the database. This should be enough.

create public rollback segment rb1 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb2 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb3 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb4 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb5 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb6 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb7 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb8 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb9 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb10 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb11 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb12 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb13 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb14 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb15 storage(initial 50K next 250K)
  tablespace rbs;
create public rollback segment rb16 storage(initial 50K next 250K)
  tablespace rbs;
```

You will run this script at the SQL*Plus prompt in "Creating a Database" on page 6-20.

Additional Information: If creating a tablespace in a raw partition, modify the data file names with a naming convention of \\.\DRIVE_LETTER: or \\.\SYMBOLIC LINK NAME. See Appendix D, "Storing Tablespaces on Raw Partitions" for more information on storing tablespaces in raw partitions.

Creating a Database

To use the BUILD_PROD.SQL script to create a database:

1. Verify that the service is started in the Windows NT Control Panel. In this example, the service name is OracleServicePROD, and its status column must display *Started*. If not, select the service name and click Start.

   You can also check the status of the service by entering the following at the MS-DOS command prompt:

   ```
   C:\> NET START
   ```

   A list of all Windows NT services currently running on the system appears. If OracleServicePROD is missing from the list, enter:

   ```
   C:\> NET START ORACLESERVICEPROD
   ```

2. Make PROD the current SID:

   ```
   C:\> SET ORACLE_SID=PROD
   ```

3. Start SQL*Plus from the MS-DOS command prompt, and connect to the database as INTERNAL:

   ```
   C:\> SQLPLUS
   ```

   ```
   SQL> CONNECT INTERNAL/PASSWORD
   ```

   The password is the one that you used to create the service, with the ORADIM -NEW command in "Creating and Starting an Oracle Service" on page 6-16.

   The message *Connected* appears.

4. Turn on spooling to save the messages:

   ```
   SQL> SPOOL BUILD_PROD.LOG
   ```

5. Run the BUILD_PROD.SQL script that you created in "Putting the CREATE DATABASE Statement in a Script" on page 6-17:

   ```
   SQL> @C:\ORACLE\ORA81\RDBMS\ADMIN\BUILD_PROD.SQL;
   ```

   If the database is created successfully, the instance is started and the message *Statement processed* appears numerous times.

If you receive any errors, there are three possible causes, as illustrated below.

Cause	Solution
The BUILD_PROD.SQL script contains syntax errors.	Correct them.
Some of the files to be created by the BUILD_PROD.SQL script already exist in the file system.	Make sure you are not using any file names already used by another database on the system.
An error occurred at the operating system level, such as a file or directory permission problem.	You should have received a series of errors in SQL*Plus, the last one of which should have the OSD- prefix. At the end of the OSD error, you typically see an operating system error number in parentheses.
	To see what the error means, do either of the following:
	From the MS-DOS command prompt, enter:
	`C:\> NET HELPMSG n`
	or
	From the SQL*Plus prompt, enter:
	`SQL> HOST NET HELPMSG n`
	where *n* is the operating system error number.

You *must* correct these problems before making another attempt to create a database.

6. Run the CATALOG.SQL script to create the data dictionary:

```
SQL> @C:\ORACLE\ORA81\RDBMS\ADMIN\CATALOG.SQL;
```

> **Note:** You may see messages such as *ORA-01432: public synonym to be dropped does not exist* while the CATALOG.SQL, CATPROC.SQL, and CATREP.SQL scripts are running. These are information messages and are intended to occur while creating a new database.
>
> If you see any unusual errors while examining the BUILD_PROD.LOG log file in step 10, see *Oracle8i Error Messages* for suggested actions.

7. Run the CATPROC.SQL script to install the objects used by the Oracle8*i* database's PL/SQL functionality:

```
SQL> @C:\ORACLE\ORA81\RDBMS\ADMIN\CATPROC.SQL;
```

8. Run the CATREP.SQL script if you want Advanced Replication functionality with the new database. Ensure that the rollback segments are large enough and are online before you run CATREP.SQL.

```
SQL> @C:\ORACLE\ORA81\RDBMS\ADMIN\CATREP.SQL;
```

9. Turn off spooling:

```
SQL> SPOOL OFF
```

10. Examine the BUILD_PROD.LOG file for any unusual errors.

> **IMPORTANT:** The new database contains two users, SYS and SYSTEM, with passwords CHANGE_ON_INSTALL and MANAGER, respectively. For security reasons, change the passwords now. Use the ALTER USER statement to change the passwords:
>
> ```
> SQL> ALTER USER SYS IDENTIFIED BY NEW_SYS_PASSWORD;
> SQL> ALTER USER SYSTEM IDENTIFIED BY NEW_SYSTEM_PASSWORD;
> ```

11. Exit SQL*Plus:

```
SQL> EXIT
```

12. Run ORADIM from the MS-DOS command prompt to set the database to start automatically when you start the computer:

```
C:\> ORADIM -EDIT -SID PROD -STARTMODE AUTO
```

Importing a Database

You can import the full export created in "Exporting an Existing Database" on page 6-12 into the new database.

You can also invoke the Import Utility, using parameter mode or interactive mode. Parameter mode is recommended, because interactive mode provides less functionality. Interactive mode exists solely for backward compatibility.

Example 6–3   Parameter Mode

```
C:\> IMP SYSTEM/PASSWORD FILE=MYEXP.DMP FULL=Y LOG=MYEXP.LOG
```

Example 6–4   Interactive Mode

```
C:\> IMP SYSTEM/PASSWORD
```

Enter only the command IMP SYSTEM/*PASSWORD* to begin an interactive session and let the Import Utility prompt you for the information it needs.

See *Oracle8i Utilities* for more information on using the Import Utility.

> **Note:** If you use parameter mode, the Import Utility considers file names and directory names to be invalid if there is a blank space. Enclose the full path in the FILE= parameter in triple quotes. For example:
>
> ```
> FILE="""C:\PROGRAM FILES\EXPORT.DMP"""
> ```
>
> or
>
> ```
> FILE="""C:\PROGRAM FILES\EXPORT FILE.DMP"""
> ```
>
> If you use the Import Utility in interactive mode, the file name or directory name can contain a space without quotes.

To import a database:

- Run the Import Utility:

  ```
  C:\> IMP SYSTEM/PASSWORD FILE=MYEXP.DMP FULL=Y LOG=MYIMP.LOG
  ```

> **IMPORTANT:** If the original database from which the export file was generated contains tablespaces that are not in the new database, the Import Utility tries to create those tablespaces with associated data files.
>
> The easy solution is to ensure that both databases contain the same tablespaces. The data files do not have to be identical. Only the tablespace names are important.

Updating the ORACLE_SID in the Registry

If this is the first database on the system or if you want to make the new database the default database, you must make a change in the registry.

1. Start the registry editor at the MS-DOS command prompt:

   ```
   C:\> REGEDT32
   ```

   The registry editor window appears.

2. Choose the \HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0 subkey for the first Oracle home on your computer. For subsequent installations to different Oracle homes on the same computer, the path is \HKEY_LOCAL_ MACHINE\SOFTWARE\ORACLE\HOME*ID,* where *ID* is the unique number identifying the Oracle home.

> **See Also:** See Appendix C, "Oracle8i Configuration Parameters
> and the Registry" for more information on the subkey locations for
> multiple Oracle homes.

3. Locate the ORACLE_SID parameter on the right side of the registry editor
window.

4. Double-click the parameter name and change the data to the new SID, which is
PROD in this example.

If you do not yet have the ORACLE_SID parameter, because this is the first
database on your system, you must create it.

To create the ORACLE_SID parameter:

1. Choose Add Value from the Edit menu.

The *Add Value* dialog box appears:
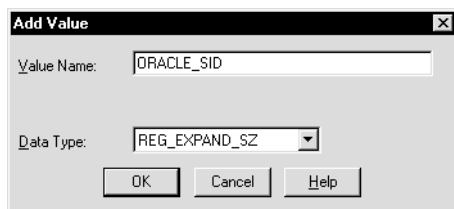
2. Enter ORACLE_SID in the Value Name text box.

3. Select REG_EXPAND_SZ (for an expandable string) in the Data Type list box.

4. Click OK.

A string editor dialog box appropriate for the data type appears:
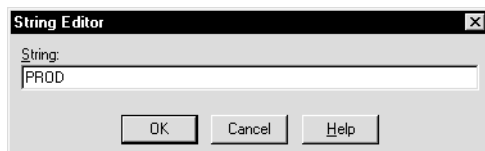
5. Enter PROD in the *String Editor* dialog box.

6. Click OK.

The registry editor adds the ORACLE_SID parameter.

7. Choose Exit from the Registry menu.

The registry exits.

Backing Up the New Database

> **WARNING:** If anything goes wrong while operating the new database without a backup, you must repeat the database creation procedure. Back up your database now to prevent loss of data.

To back up the new database:

1. Shut down the database instance and stop the service:

```
C:\> ORADIM -SHUTDOWN -SID PROD -USRPWD PASSWORD

-SHUTTYPE SRVC,INST -SHUTMODE I
```

> **WARNING:** Although ORADIM returns the prompt immediately, you must wait for the database and the service to stop completely before continuing to Step 2. Wait until the Control Panel indicates the OracleServicePROD service has stopped. If you do not do this, the backup may be useless as it was taken while data was being written to the data files.

2. Using the tool of your choice, back up the database files.

Database files consist of the initialization parameter file, control files, online redo log files, and data files.

When the backup is complete, you can start the database again, create users and objects, if necessary, make any other changes, and use the database.

Be sure to back up the database backup after making any significant changes to the database, such as switching the ARCHIVELOG mode, or adding a tablespace or data file.

> **See Also:** Chapter 11, "Backing Up and Recovering Database Files", *Oracle8i Concepts*, *Oracle8i Backup and Recovery Guide*, and *Oracle8i Administrator's Guide* for more information on archiving and backup/recovery.

> **Warning:** Do not store database files on a compressed drive. This can result in write errors and decreased performance.

Using ORADIM to Administer an Oracle Instance

ORADIM is a command line tool that is only available with the Oracle8*i* database.

You only need to use ORADIM if you are manually creating, deleting, or modifying databases. Oracle Database Configuration Assistant is an easier tool to use for this purpose.

ORADIM and Oracle Database Configuration Assistant perform similar tasks. The following table compares what you can do with these tools.

ORADIM	Oracle Database Configuration Assistant
■ Can create, start, stop, modify, and delete instances (and not any associated database files) at the MS-DOS command prompt.	■ Can create and delete databases (and their associated instances and services). Note that the assistant cannot start or stop existing databases.
■ Can be used to modify an instance. You can modify an existing instance to change such values as the instance name, the password, the startup mode, or the shutdown mode.	■ Cannot be used to modify an instance.
■ Only creates the password file and the related service. The database (that is, the database files) is not created.	■ Creates the database, the associated instance, the service, and the password file.

When you use ORADIM, a log file called ORADIM.LOG opens in *ORACLE_BASE\ORACLE_HOME*\DATABASE, or in the directory specified by the ORA_CWD registry parameter. All operations, whether successful and failed, are logged in this file. You must check this file to verify the success of an operation.

The following sections describe the ORADIM commands and parameters. Note that each command is preceded by a dash (-).

To get a list of ORADIM parameters and descriptions...

Use this syntax:	`ORADIM -?	-H	-HELP` **Note:** Specifying ORADIM without any options also returns a list of ORADIM parameters and descriptions.
Example:	`C:\> ORADIM -?`

Creating an Instance

To create an instance...

Use this syntax:	`ORADIM -NEW -SID SID	-SRVC SERVICE_NAME [-INTPWD INTERNAL_PWD] -SHUTTYPE SRVC	INST	SRVC, INST`  `[-MAXUSERS NUMBER][-STARTMODE AUTO	MANUAL][-PFILE FILENAME]`
Example to create an instance called PROD:	`C:\> ORADIM -NEW -SID PROD -INTPWD MYPASSWORD1 -STARTMODE AUTO` `-PFILE C:\ORACLE\ADMIN\PROD\PFILE\INIT.ORA`
Syntax description:	■ -NEW — Indicates that you want to create a new instance. This is a mandatory parameter.
	■ -SID *SID* — The name of the instance you want to create. You must specify either this parameter or the -SRVC parameter described below.
	■ -SRVC *SERVICE_NAME* — The name of the service you want to create (OracleService*SID*). You must specify either this parameter or the -SID parameter described above.
	■ -INTPWD *INTERNAL_PWD* — The password for the INTERNAL account. The -INTPWD option is not required. If you do not specify it, operating system authentication is used, and no password is required. See "Automatically Enabling Operating System Authentication During Installation" on page 8-13 for a description of features.
	■ -MAXUSERS *NUMBER* — The number of users defined in the password file. The default is 5.
	■ -STARTMODE *AUTO, MANUAL* — Indicates whether to start the instance automatically or manually at startup. The default setting is MANUAL.
	■ -PFILE *FILENAME* — The INIT.ORA file to be used with this instance. Ensure that you specify the complete pathname of this file, including drive letter.
	■ -SHUTTYPE *SRVC, INST* — Indicates whether to stop the service or the instance. Both can be specified. This is a mandatory parameter.

Starting an Instance

To start an instance...		
Use this syntax:	ORADIM –STARTUP –SID *SID* [–USRPWD *USER_PWD*] [–STARTTYPE SRVC │ INST │ SRVC, INST] [–PFILE *FILENAME*]	
Example to start an instance called PUMA:	C:\> ORADIM –STARTUP –SID PUMA –STARTTYPE SRVC –PFILE C:\ORACLE\ADMIN\PROD\PFILE\INIT.ORA	
Syntax description:	■ -STARTUP	Indicates that you want to start an instance that already exists. This is a mandatory parameter.
	■ -SID *SID*	The name of the instance you want to start. This is a mandatory parameter.
	■ -USERPWD *USER_ PWD*	The password.
	■ -STARTTYPE *SRVC, INST*	Indicates whether to start the service or the instance. One or both values can be specified. If not specified, the registry is checked for the current setting.

Stopping an Instance

To stop an instance:		
Use this syntax:	ORADIM –SHUTDOWN –SID *SID* [–USRPWD *USER_PWD*] [–SHUTTYPE SRVC │ INST │ SRVC, INST] [–SHUTMODE A │ I │ N]	
Example to stop an instance called PUMA:	C:\> ORADIM –SHUTDOWN –SID PUMA –SHUTTYPE SRVC INST	
Syntax description:	■ -SHUTDOWN	Indicates that you want to stop an instance. This is a mandatory parameter.
	■ -SID *SID*	The name of the instance you want to stop. This is a mandatory parameter.
	■ -USERPWD *USER_ PWD*	The password.
	■ -SHUTTYPE *SRVC, INST*	Indicates whether to stop the service or the instance. One or both values can be specified. If not specified, the registry is checked for the current setting.

	-SHUTMODE A, I, N	Specifications on how to stop an instance; A indicates abort mode, I indicates immediate mode, and N indicates normal mode. This is an optional parameter. If you do not specify how to stop an instance, normal is the default mode.

Modifying an Instance

To modify an instance...	
Use this syntax:	`ORADIM –EDIT –SID SID [–NEWSID NEWSID] [–INTPWD INTERNAL_PWD]` `[–STARTMODE AUTO	MANUAL][–PFILE FILENAME]`
Example to modify an instance called PROD:	`C:\> ORADIM –EDIT –SID PROD –NEWSID LYNX –INTPWD MYCAT123 –STARTMODE` `AUTO –PFILE C:\ORACLE\ADMIN\LYNX\PFILE\INIT.ORA`
Note:	You can modify an existing instance, in this example PROD, to change such values as the instance name, the password, the startup mode, and the number of users.
Syntax description:	■ -EDIT — Indicates that you want to modify an instance. This is a mandatory parameter.
	■ -SID *SID* — The name of the instance you want to modify. This is a mandatory parameter.
	■ -NEWSID *NEWSID* — The new instance name. This is an optional parameter.
	■ -INTPWD *INTERNAL_PWD* — The password for the INTERNAL account. **Note:** This parameter cannot be used to change the password, as it does not overwrite the existing password file. It can only create a new password file when none already exists. To create a new password file, use ORAPWD, or delete the Oracle8*i* services (this action implicitly deletes the associated password file) and then recreate the Oracle8*i* services (this action implicitly creates the associated password file). See "Password Utility (ORAPWD)" in Chapter 2, "Database Tools Overview".
	■ -STARTMODE AUTO, MANUAL — Indicates whether to start the instance automatically or manually at startup. The default setting is MANUAL.
	■ -PFILE *FILENAME* — The INIT.ORA file to be used with this instance. Ensure that you specify the complete pathname of this file, including drive letter.

7

Administering a Database

This chapter describes how to administer Oracle8*i* for Windows NT.

Specific topics discussed are:

- Managing Oracle Services
- Starting and Shutting Down a Database with SQL*Plus
- Starting and Shutting Down a Database Using Services
- Running Multiple Instances
- Creating Password Files
- Deleting Password Files
- Connecting as INTERNAL with a Password File
- Connecting Remotely to the Database as SYS or INTERNAL
- Changing the INTERNAL Password
- Encrypting Database Passwords
- Creating Control, Data, and Log Files on Remote Computers
- Archiving Redo Log Files
- Using the ORADEBUG Utility

Managing Oracle Services

This section provides information on the following:

- Oracle Service Naming Conventions for Multiple Oracle Homes
- Available Oracle Services
- Starting Oracle Services
- Stopping Oracle Services
- Auto-starting Oracle Services

Oracle Service Naming Conventions for Multiple Oracle Homes

Oracle8*i* for Windows NT allows you to have multiple, active Oracle home directories on a single computer. Chapter 3, "Multiple Oracle Homes and Optimal Flexible Architecture" describes this feature. Multiple Oracle homes affect the naming conventions for Oracle services. As you perform installations into Oracle home directories:

- you must accept the default Oracle *home name* provided or specify a different name for each Oracle home directory during installation, which is added to most service names
- you will be prompted to identify a system identifier (SID) and database name for each installation

These two figures show how the *Services* dialog box appears with two Oracle8*i* databases on a single computer:

Service names for first Oracle8*i* database installation. A home name appears in most service names(OraHome81).

Service names for second Oracle8*i* database installation. Home names may be specified by the user.

SID for the Oracle8*i* database installation as entered by the user will appear in the service name.
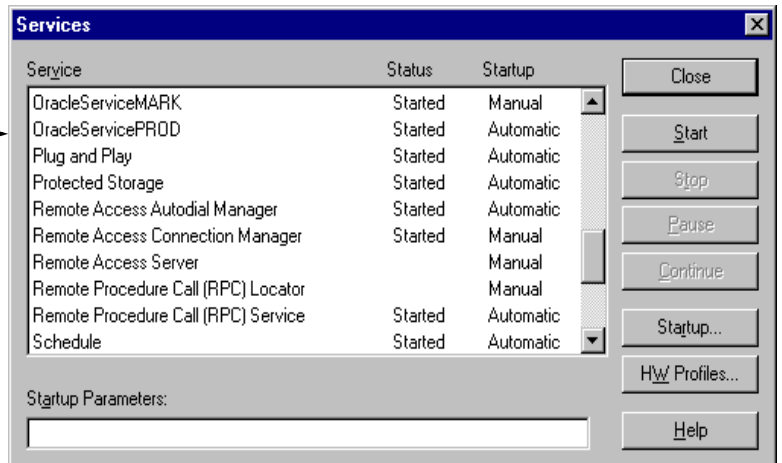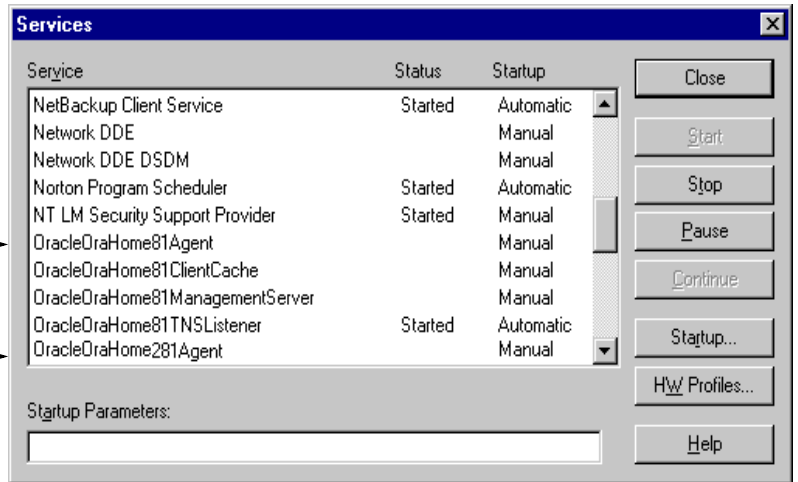
Available Oracle Services

Depending on the products that you have installed, a number of Oracle services are started when you restart your Windows NT computer. The two main Oracle services are:

Service Name	Description
OracleService*SID*	Created for the database instance *SID*. An Oracle instance is a logical term that refers to: ■ an Oracle service called OracleService*SID* ■ a database Each Oracle instance must have a system identifier (SID). A SID is a unique name for an Oracle database instance that can be up to 64 alphanumeric characters in length. For example, if the SID for the Oracle8*i* database is ORCL, it is appended to the service OracleService. The instance name is the same as the value of the ORACLE_SID registry configuration parameter.
Oracle*HOME_NAME*TNSListener	Listens for and accepts incoming connection requests from client applications. Automatically starts when the Windows NT computer restarts. The *HOME_NAME* that displays in this service name is the value you entered in the *Name* field on the *File Locations* dialog box of Oracle Universal Installer.

Additional Oracle services related to specific products or database features are also available:

Service Name	Description
Oracle*HOME_NAME*Agent	Listens for and responds to job and event requests sent from the Oracle Enterprise Manager console.
OracleWebAssistant0[1]	Enables information from database queries to be published to a Web page at specified time intervals.
OracleMTSService0[1]	Provides the COM communication interface between Microsoft Transaction Server (and its MS DTC component) and the Oracle8*i* database.

[1] This is the name for this service if you have only one Oracle home directory on your computer. Each additional Oracle home directory on your computer uses the naming convention OracleWebAssistant1, OracleWebAssistant2, OracleMTSService1, OracleMTSService2, and so on for this service.

Services for network products are also available:

Service Name	Description
■ Oracle*HOME_NAME*TNSListener or, for example: ■ Oracle*HOME_NAME*TNSListener*LSNR*	The service name, if you are using the default network listener name LISTENER. ■ *LSNR* is the non-default network listener name. It is only created if the following command has been run: `LSNRCTL START LSNR`
Oracle*HOME_NAME*ClientCache	Used for the Client Cache Service.
Oracle*HOME_NAME*CMAdmin Oracle*HOME_NAME*CMan	Used for the product Oracle Connection Manager.

> **See Also:** *Net8 Administrator's Guide* for general information on Oracle network services.

Starting Oracle Services

Oracle services must be started for you to use the Oracle8*i* database and its products.

Start Oracle services in either of three ways:

- From the Control Panel
- From the MS-DOS command prompt
- From the Oracle Administration Assistant for Windows NT

> **Note:** You can start the Oracle8*i* database when you start OracleService*SID*. See "Starting and Shutting Down a Database Using Services" on page 7-12, for information on registry parameters that enable you to do this.

To start Oracle Services from the Control Panel:

1. Choose Start > Settings > Control Panel.

   The *Control Panel* window appears.

2. Double-click Services.

   The *Services* dialog box appears.

3. Find the service you want to start in the list, and verify that it has a status of Started. If it does not, select it and choose Start.

4. Click Close to exit the *Services* dialog box.

To start Oracle Services from the MS-DOS command prompt:

1. Enter the following command to start an Oracle service at the MS-DOS command prompt:

   ```
   C:\> NET START SERVICE
   ```

   where *SERVICE* is a specific service name, such as OracleServiceORCL.

If you cannot find OracleService*SID* in the list, use ORADIM to create it. See "Using ORADIM to Administer an Oracle Instance" on page 6-26 for instructions on using ORADIM.

To start Oracle Services from the Oracle Administration Assistant for Windows NT:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration> Oracle Administration Assistant for Windows NT.

2. Right-click the *SID*.

   where *SID* is a specific instance name, such as ORCL.

3. Choose Start Service

   where *SERVICE* is a specific service name, such as OracleServiceORCL.

Stopping Oracle Services

On occasion (for example, if you want to re-install the Oracle8*i* database), you must stop Oracle services. Stop Oracle services in either of three ways:

- From the Control Panel

- From the MS-DOS command prompt

- From the Oracle Administration Assistant

  > **Note:** You can stop the Oracle8*i* database in normal, immediate, or abort mode when you stop OracleService*SID*. See "Starting and Shutting Down a Database Using Services" on page 7-12 for information on registry parameters that enable you to do this.

To stop Oracle8*i* Services from the Control Panel:

1. Choose Start > Settings > Control Panel.

The *Control Panel* window appears.

2. Double-click Services.

The *Services* dialog box appears.

3. Select Oracle*HOME_NAME*TNSListener and choose Stop.

Oracle*HOME_NAME*TNSListener is stopped.

4. Select OracleService*SID* and choose Stop.

5. Click OK.

OracleService*SID* is stopped.

To stop Oracle Services from the MS-DOS command prompt:

1. Enter the following command to stop an Oracle service at the MS-DOS command prompt:

```
C:\> NET STOP SERVICE
```

where *SERVICE* is a specific service name, such as OracleServiceORCL.

To stop Oracle Services from the Oracle Administration Assistant for Windows NT:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration> Oracle Administration Assistant for Windows NT.

2. Right-click the *SID*

where *SID* is a specific instance name, such as ORCL.

3. Choose Stop Service

where *SERVICE* is a specific service name, such as OracleServiceORCL.

Auto-starting Oracle Services

You can start Oracle services whenever the Windows NT computer is restarted. Use the *Service* dialog box to configure when and how the Oracle8*i* database is started.

To automatically start the Oracle8*i* database each time you restart:

1. Choose Start > Settings > Control Panel.

   The *Control Panel* window appears.

2. Double-click Services.

   The *Services* dialog box appears.

3. Select the service OracleService*SID* and choose the Startup button.

   The *Service* dialog box appears.

4. Choose Automatic from the Startup Type field.

5. Click OK.

6. Click Close to exit the *Services* dialog box.

To automatically start Oracle Services from the Oracle Administration Assistant for Windows NT:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration> Oracle Administration Assistant for Windows NT.

2. Right-click the *SID*

   where *SID* is a specific instance name, such as ORCL.

3. Choose Startup/Shutdown Options.

4. Click the Oracle NT Service tab.

5. In the Oracle NT Service Startup Type box click Automatic.
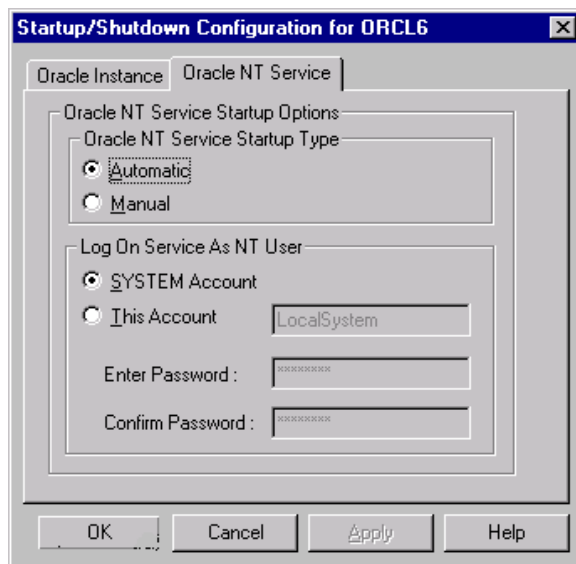
Starting and Shutting Down a Database with SQL*Plus

These instructions assume that a database instance is created and the services are started.

> **Note:** The directory path examples in this chapter follow Optimal Flexible Architecture (OFA) guidelines (for example, *ORACLE_BASE**ORACLE_HOME*\\RDBMS\\ADMIN). If you specified non-OFA compliant directories during installation, your directory paths will differ. See section "OFA and Multiple Oracle Home Configurations" on page 3-23 for information.

To start or shut down an Oracle8*i* database:

1. Go to your Oracle8*i* database server.

2. Start SQL*Plus at the MS-DOS command prompt:

    ```
    C:\> SQLPLUS
    ```

3. Connect to the Oracle8*i* database with the INTERNAL user name:

    ```
    SQL> CONNECT INTERNAL
    ```

4. Follow the instructions below:

If You Want to...	Then Enter...
Start a database with the default parameter file	`SQL> STARTUP`
	This command uses the default INIT.ORA file located in the *ORACLE_BASE*\\ADMIN*DB_NAME*\\PFILE directory.
Start a database with a file other than the default parameter file	`SQL> STARTUP PFILE=PATH\FILENAME`
	This command uses the INIT.ORA file specified in *PATH\FILENAME*. This example starts the database using a file named INIT2.ORA in C:\\ORA81\\ADMIN\\ORCL\\PFILE:
	`SQL> STARTUP PFILE=C:\ORA81\ADMIN\ORCL\PFILE\INIT2.ORA`

If You Want to...	Then Enter...
Stop the database	`SQL> SHUTDOWN [MODE]` where *MODE* is one of the following: ■　Normal The database waits for all currently-connected users to disconnect and disallows any new connections before shutting down. This is the *default* mode. ■　Immediate The database terminates and rolls back active transactions, disconnects clients, and shuts down. ■　Abort The database terminates active transactions and disconnects users; it does not roll back transactions. The database performs automatic recovery and rollback the next time it is started. Use this mode only in emergencies.

> **Note:** See "Choosing a Database Tool" on page 2-2 for a list of other tools that can start the database and the *Oracle8i Administrator's Guide* for information on options you can specify when starting your database.

Starting and Shutting Down a Database Using Services

You can start or shut down the Oracle8*i* database by starting or stopping the service OracleService*SID* in the Control Panel. This automated procedure is equivalent to manually entering the following:

If You...	These Commands are Performed...
Start OracleService*SID*	`C:\> ORADIM - STARTUP -SID SID -USERPWD PASSWORD -STARTTYPE SRV,INST`
Stop OracleService*SID*	`C:\> ORADIM - SHUTDOWN -SID SID -USERPWD PASSWORD -SHUTTYPE SRV,INST -SHUTMODE -I`

To start the database by starting OracleService*SID*:

1.　Review or set the following registry parameters. Note that ORADIM, when used to create or edit instances, automatically sets these values in the registry.

Parameter	When Set to TRUE, this Parameter...
ORA_*SID*_AUTOSTART	When set to TRUE (the default value), starts the database when OracleService*SID* is started.
ORA_*SID*_PFILE	Sets the full path to the INIT.ORA parameter file. The default path is *ORACLE_BASE*\ADMIN*DB_NAME*\PFILE\INIT.ORA.

See Appendix C, "Oracle8i Configuration Parameters and the Registry" for instructions on adding and editing registry parameters.

The exact location in which to set these parameters is determined by the number of Oracle home directories on your computer:

If You Have...	Then Add These Parameters in...
One home directory	HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0
Additional directories	HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID*
	where *ID* is incremented for each additional Oracle home directory on your computer.

2. Choose Start > Settings > Control Panel.

   The *Control Panel* window appears.

3. Double-click Services.

   The *Services* dialog box appears.

4. Select OracleService*SID* and choose Start.

   This automatically starts ORADIM and issues the -STARTUP command using the initialization parameter file identified by ORA_*SID*_PFILE.

To shut down the database by stopping OracleService*SID*:

1. Set either of the following parameters to TRUE in the registry:

Parameter	When Set to TRUE, this Parameter...
ORA_SHUTDOWN	Enables the selected Oracle8*i* database to be shut down. This includes any database in the current Oracle home.
ORA_*SID*_SHUTDOWN	Shuts down the Oracle8*i* database identified by the *SID* value.

If either is set to FALSE (the default setting), it is possible to shut down the database by stopping OracleService*SID*, but it's not recommended because it results in an abnormal shutdown of the database.

The exact location in which to set these parameters is determined by the number of Oracle home directories on your computer:

If You Have...	These Parameters are Located in...
One home directory	HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0
Two or more home directories	HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID* where *ID* is incremented for each additional Oracle home directory on your computer.

Additional Information: See Appendix C, "Oracle8i Configuration Parameters and the Registry" for instructions on adding and editing registry parameters.

2. Set the following optional parameters to appropriate values in the registry:

Parameter	Description
ORA_*SID*_ SHUTDOWNTYPE	Set to A (abort), I (immediate), or N (normal) to indicate the database shutdown mode. The default mode is I (immediate) if you do not set this parameter.
ORA_*SID*_ SHUTDOWN_ TIMEOUT	Sets the maximum time to wait before the service for a particular SID stops.

3. Choose Start > Settings > Control Panel.

   The *Control Panel* window appears.

4. Double-click Services.

   The *Services* dialog box appears.

5. Select OracleService*SID* and choose Stop.

   This automatically starts ORADIM, which issues the -SHUTDOWN command in the mode indicated by ORA_*SID*_SHUTDOWNTYPE, and shuts down your Oracle8*i* database.

To start or stop a database using Oracle Services from the Oracle Administration Assistant for Windows NT:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration> Oracle Administration Assistant for Windows NT.

2. Right-click the *SID*

   where *SID* is a specific instance name, such as ORCL.

3. Choose Startup/Shutdown Options.

4. Click the Oracle Instance tab.

5. Select Start up instance when service is started or select Shut down instance when service is stopped.
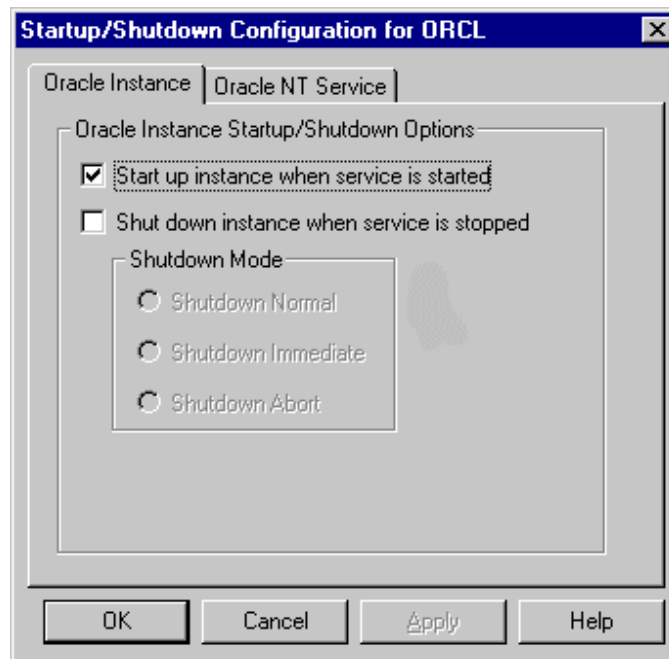
Running Multiple Instances

To run multiple instances, ensure that you have already created each instance and started the services for each instance using ORADIM or the *Services* dialog box in

Windows NT. You then run multiple instances by starting each of the instances using SQL*Plus.

To run multiple instances:

1. Ensure that you have already created each instance.

2. Ensure that you started the services for each instance using ORADIM or the *Services* dialog box of the Windows NT Control Panel.

3. Set the ORACLE_SID configuration parameter at the MS-DOS command prompt to the SID for each instance you want to run:

    ```
    C:\> SET ORACLE_SID=SID
    ```

    where *SID* is the name of the Oracle8*i* database instance.

4. Start SQL*Plus:

    ```
    C:\> SQLPLUS
    ```

5. Connect as INTERNAL:

    ```
    SQL> CONNECT INTERNAL
    ```

6. Start up the database with the new instance:

    ```
    SQL> STARTUP PFILE=ORACLE_BASE\ADMIN\DB_NAME\PFILE\INIT.ORA
    ```

    where *ORACLE_BASE* is C:\ORACLE by default (unless you changed it during installation) and *DB_NAME* is the name of the instance.

Creating Password Files

Use the Password Utility ORAPWD to create password files. ORAPWD is automatically installed with the Oracle8*i* Utilities. Password files are located in the *ORACLE_BASE\ORACLE_HOME*\DATABASE directory and are named PWD*SID*.ORA, where *SID* identifies the Oracle8*i* database instance. Password files can be used for local or remote connections to an Oracle8*i* database. The example below describes how to perform a local connection.

To create a password file:

1. Create a password file with ORAPWD:

```
C:\> ORAPWD FILE=PWDSID.ORA PASSWORD=PASSWORD ENTRIES=MAX_USERS
```

The essential elements of a password file are:

Element	Description
SID	Identifies the database instance.
FILE	Specifies the password file name.
PASSWORD	Sets the password for the INTERNAL and SYS accounts.
ENTRIES	Sets the maximum number of entries in the password file. This corresponds to the maximum number of distinct users allowed to connect to the database with the SYSDBA and SYSOPER DBA privileges simultaneously.

2. Set the INIT.ORA file parameter REMOTE_LOGIN_PASSWORDFILE to EXCLUSIVE or SHARED. Definitions for all possible values are described below:

Element	Description
EXCLUSIVE	Specifies that only one instance can use the password file and that the password file contains names other than SYS and INTERNAL. Oracle8*i* looks in the registry for the value of the ORA_*SID*_PWFILE parameter. If a value is unspecified, it looks in the registry for the value of the ORA_PWFILE parameter, which points to a file containing the INTERNAL password as well as user names, passwords, and privileges. If that is not set, it uses the default of *ORACLE_BASE**ORACLE_HOME*\\DATABASE\\PWD*SID*.ORA.
SHARED	Specifies that multiple instances can use the password file (for example, a parallel server environment). However, the only users recognized by the password file are SYS and INTERNAL. You cannot log in with SYSOPER or SYSDBA privileges even if those privileges are granted in the password file. The SHARED value of this parameter affords backward compatibility with earlier releases. The Oracle8*i* database looks for the same files as it does when the value is EXCLUSIVE, but only the INTERNAL account is available for privileged access. This is the default value.
NONE	Specifies that the Oracle8*i* database ignores the password file and that privileged users are authenticated by the Windows NT operating system. NONE is the default setting.

3. Start SQL*Plus:

```
C:\> SQLPLUS
```

4. Connect as INTERNAL:

```
SQL> CONNECT INTERNAL
```

5. Start the Oracle8*i* database:

```
SQL> STARTUP
```

6. Grant appropriate privileges to each user who needs to perform database administration. For example:

```
SQL> GRANT SYSDBA TO SCOTT;
```

If successful, the following message displays:

```
Statement Processed.
```

This adds SCOTT to the password file and enables SCOTT to connect to the database with SYSDBA privileges. Use SQL*Plus to add or delete user names, user passwords, and user privileges in password files.

7. Connect to the Oracle8*i* database with DBA privileges for SCOTT:

```
SQL> CONNECT SCOTT/TIGER AS SYSDBA
```

You are connected to the Oracle8*i* database.

Caution: Copying or manually moving password files may result in ORADIM not being able to find a password to start an instance.

Viewing Password Files

The password file is automatically hidden. This section describes two ways of viewing the password file:

- To see the password file from the MS-DOS command prompt:

- To see the password file from Windows NT Explorer:

To see the password file from the MS-DOS command prompt:

To...	Enter...
See the password file	`C:\ORACLE\ORA81\DATABASE> ATTRIB`
	The password file displays:
	`A   H   C:\ORACLE\ORA81\DATABASE\PWDSID.ORA`
Make the password file visible	`C:\ORACLE\ORA81\DATABASE> ATTRIB -H PWDSID.ORA`
	Note: The password file must be visible before you can move or copy it.
Hide the password file again	`C:\ORACLE\ORA81\DATABASE> ATTRIB +H PWDSID.ORA`

To see the password file from Windows NT Explorer:

1. Open Windows NT Explorer.

2. Open the folder to view.

3. Click Folder Options from the View main menu.

4. Click the View tab.

5. Follow the instructions below:

To...	Click...
See the password file	Show all files.
Hide the password file	Do not show hidden files.

Deleting Password Files

This section describes how to delete a password file.

To delete a password file:

1. Make the password file visible at the MS-DOS command prompt or in Windows NT Explorer by following the instructions in section "Viewing Password Files" on page 7-18.

2. Delete the password file based on whether you made it visible at the MS-DOS command prompt or in Windows NT Explorer.

Connecting as INTERNAL with a Password File

You can connect as INTERNAL with a password file. The password for INTERNAL is ORACLE if you installed your database through the Typical or Minimal installation types.

To connect as INTERNAL with a password file:

1. Follow the procedures in "Creating Password Files" on page 7-16 to create a password file with ORAPWD.

2. Set the INIT.ORA file parameter REMOTE_LOGIN_PASSWORDFILE to EXCLUSIVE or SHARED.

> **Note:** When REMOTE_LOGIN_PASSWORDFILE is set to NONE, connecting remotely to a database as INTERNAL is prohibited even if the correct password is supplied.

3. Connect to your Oracle8*i* database as follows:

   ```
   SQL> CONNECT INTERNAL/PASSWORD
   ```

   where *PASSWORD* is the password created with ORAPWD or ORADIM.

Connecting Remotely to the Database as SYS or INTERNAL

When connecting to the starter database from a remote machine as SYS or INTERNAL, you must use a different password from the one described in Chapter 6 of the *Oracle8i Installation Guide for Windows NT* when logging on with SYSDBA privileges. This is because the password file enables database access in this situation and it requires the password ORACLE for this purpose.

Changing the INTERNAL Password

Change the INTERNAL user account password with either ORADIM or ORAPWD.

To change the password with ORADIM:

1. Delete the SID for the password you want to change:

   ```
   C:\> ORADIM -DELETE -SID SID
   ```

   where *SID* is the SID to delete.

2. Verify that the PWD*SID*.ORA and STRT*SID*.CMD files have been removed from the *ORACLE_HOME*\\DATABASE\\ directory.

3. Create the same SID again and specify a new INTERNAL password:

```
C:\> ORADIM –NEW –SID SID –INTPWD NEW_PASSWORD – STARTMODE AUTO – MAXUSERS N
–PFILE <drive: \path\init{sid}.ora>
```

where *SID* is the same SID to recreate, *NEW_PASSWORD* is the new INTERNAL password, and *N* is the maximum number of DBAs/operators who can be logged in at once with this password.

To change the password with ORAPWD:

1. See section "Deleting Password Files" on page 7-19 for instructions on deleting the password file.

2. See section "Creating Password Files" on page 7-16 for instructions on creating a password file.

To change the password with SQL*Plus:

The section assumes the password file is already created and the INIT.ORA file parameter REMOTE_LOGIN_PASSWORDFILE is set to SHARED or EXCLUSIVE.

1. Start SQL*Plus:

```
C:\> SQLPLUS
```

2. Connect with the INTERNAL user name:

```
SQL> CONNECT SYS/password
```

3. Change the password for the SYS user name, for which INTERNAL is an alias:

```
SQL> ALTER USER SYS IDENTIFIED BY NEW_PASSWORD;
```

Encrypting Database Passwords

With the Oracle8*i* database, you can encrypt the password used to verify a remote database connection.

To enable password encryption:

1. Add DBLINK_ENCRYPT_LOGIN to the initialization parameter INIT.ORA file on the server computer.

2. Set DBLINK_ENCRYPT_LOGIN equal to TRUE.

3. Set the ORA_ENCRYPT_LOGIN configuration variable on the client computer to TRUE. See Appendix C, "Oracle8i Configuration Parameters and the Registry" for instructions on adding and setting configuration parameters in the registry.

Once these parameters are set to TRUE, whenever a user attempts a remote login, the Oracle8*i* database encrypts the password before sending it to the remote database. If the connection fails, the failure is noted in the audit log. The Oracle8*i* database then checks if either of these parameters is set to FALSE. If so, the Oracle8*i* database attempts the connection again using an unencrypted version of the password. If the connection is successful, the success is noted in the audit log, and the connection proceeds.

> **Note:** Releases prior to release 7.1 do not support encrypted passwords. If you are connecting to an earlier version of the Oracle database, you must set the initialization parameter DBLINK_ENCRYPT_LOGIN to FALSE for the connection to succeed.

Creating Control, Data, and Log Files on Remote Computers

Although it is possible for Oracle to access database files on remote computers using Universal Naming Convention (UNC), it is not recommended or supported because of performance and network reliability concerns.

UNC is a PC format for specifying the location of resources on a local area network. UNC uses the following format:

```
\\server-name\shared-resource-pathname
```

For example, to access the file SYSTEM01.DBF in the directory C:\ORACLE\ORA81\ORADATA\ORCL on the shared server ARGON, you reference the file as:

```
\\ARGON\ORACLE\ORA81\ORADATA\ORCL\SYSTEM01.DBF
```

Note that the location of archive log files cannot be specified using UNC. If you set the LOG_ARCHIVE_DEST_*n* initialization parameter to a UNC specification, the database does not start and you receive the following errors:

```
ORA-00256: error occurred in translating archive text string '\meldell\rmdrive'
ORA-09291: sksachk: invalid device specified for archive destination
OSD-04018: Unable to access the specified directory or device
O/S-Error: (OS 2) The system cannot find the file specified
```

Ensure that you set the LOG_ARCHIVE_DEST_*n* initialization parameter to a mapped drive.

> **Note:** An ORA-00256 error also occurs if you enter:
>
> ```
> \\\meldell\rmdrive
> ```
>
> or
>
> ```
> \\\meldell\\rmdrive
> ```
>
> Control files required the additional backslashes for release 8.0.4, but redo files and data files did not.

Archiving Redo Log Files

Your Oracle8*i* database is created in NOARCHIVELOG mode if you installed your database through the Typical or Minimal installation. If you created your databas through the Custom option of Database Configuration Assistant, you had the choice of either ARCHIVELOG or NOARCHIVELOG.

In NOARCHIVELOG mode, redo logs are not archived. This protects the database from instance failure, but not from disk failure. Setting your archive mode to ARCHIVELOG and enabling automatic archiving causes redo log files to be archived. This protects the database from both instance and disk failure.

This section describes how to change the archive mode to ARCHIVELOG and enable automatic archiving. See the chapter "Archiving Redo Information" of the *Oracle8i Administrator's Guide* for complete descriptions of the ARCHIVELOG and NOARCHIVELOG modes.

Step 1: Change the Archive Mode to ARCHIVELOG

To change the archive mode to ARCHIVELOG:

1. Start SQL*Plus at the MS-DOS command prompt:

   ```
   C:\> SQLPLUS
   ```

2. Connect to the Oracle8*i* database with the INTERNAL user name:

   ```
   SQL> CONNECT INTERNAL
   ```

3. If the database is open, shut it down:

   ```
   SQL> SHUTDOWN
   ```

4. Mount the database:

   ```
   SQL> STARTUP MOUNT
   ```

5. Enter the following command:

   ```
   SQL> ARCHIVE LOG LIST
   ```

   The following output indicates the database is not in archive mode:

   ```
   Database log mode              No Archive Mode
   Automatic archival             Disabled
   Archive destination            %RDBMS%\
   Oldest online log sequence     34
   Current log sequence           37
   ```

6. Change the archive mode to ARCHIVELOG:

   ```
   SQL> ALTER DATABASE ARCHIVELOG;
   ```

7. Enter the following command:

   ```
   SQL> ARCHIVE LOG LIST
   ```

   The following output indicates the database is now in archive mode:

   ```
   Database log mode              Archive Mode
   Automatic archival             Disabled
   Archive destination            %RDBMS%\
   ```

```
Oldest online log sequence       34

Current log sequence             37
```

8. Open the database:

```
SQL> ALTER DATABASE OPEN;
```

9. Continue to "Step 2: Enable Automatic Archiving" on page 7-25.

Step 2: Enable Automatic Archiving

To enable automatic archiving:

1. Open the *ORACLE_BASE*\ADMIN\ *DB_NAME*\INIT.ORA file.

2. Find the following three parameters:

```
# LOG_ARCHIVE_START = TRUE

# LOG_ARCHIVE_DEST_1 = %ORACLE_HOME%\DATABASE\ARCHIVE

# LOG_ARCHIVE_FORMAT = "%%ORACLE_SID%%T%TS%S.ARC"
```

3. Remove the # sign from in front of each.

> **Note:** The double quotes around LOG_ARCHIVE_FORMAT do not need to be removed.

4. Edit the LOG_ARCHIVE_DEST_*n* value to identify an existing drive and directory in which to archive your filled redo logs.

5. Edit the LOG_ARCHIVE_FORMAT value to indicate the appropriate archiving format:

Format	Description	Example
%%ORACLE_SID%%%T.ARC	Specifies the thread number. This number is padded to the left by zeroes. The default value is one with a range of up to three characters.	*SID*0001.ARC
%%ORACLE_SID%%%S.ARC	Specifies the log sequence number. This number is padded to the left by zeroes. The default value is one with a range of up to five characters.	*SID*0001.ARC

Format	Description	Example
%%ORACLE_SID%%%t.ARC	Specifies the thread number. The number is not padded. The default value is one with no range limit on characters.	*SID*1.ARC
%%ORACLE_SID%%%s.ARC	Specifies the log sequence number. The number is not padded. The default value is one with no range limit on characters.	*SID*1.ARC

6. Save your changes.

7. Exit the file.

8. Shut down the database:

```
SQL> SHUTDOWN
```

9. Restart the database

```
SQL> STARTUP
```

10. Enter the following command:

```
SQL> ARCHIVE LOG LIST
```

The following output indicates that automatic archiving of redo log files is enabled and an archiving destination is specified:

```
Database log mode          Archive Mode

Automatic archival         Enabled

Archive destination        C:\BACKUP

Oldest online log sequence    34

Current log sequence          37
```

Using the ORADEBUG Utility

The ORADEBUG utility is a debugging tool that sends debug commands through SQL*Plus to Oracle processes. It is primarily for use by developers and Oracle Support Services personnel. Only use this utility when instructed to do so by Oracle Support Services. Note that you must have database administrator privileges to use ORADEBUG.

To start ORADEBUG:

1. Start SQL*Plus from the MS-DOS command prompt and connect to the database as INTERNAL. For example:

```
C:\> SQLPLUS
SQL> CONNECT INTERNAL
```

2. Enter the following at the SQL*Plus prompt:

```
SQL> ORADEBUG
```

ORADEBUG runs and prompts you for parameters. To obtain a list of these parameters, enter the following at the SQL*Plus prompt:

```
SQL> ORADEBUG HELP
```

The output from most debug commands is written to a trace file. Trace files are created in the directory specified by the INIT.ORA initialization parameters BACKGROUND_DUMP_DEST and USER_DUMP_DEST. By default, these parameters are set to *ORACLE_BASE*\ADMIN*DB_NAME*\BDUMP and *ORACLE_BASE*\ADMIN*DB_NAME*\UDUMP, respectively. If you want to find the location of your trace file, enter the following at the SQL*Plus prompt:

```
SQL> ORADEBUG TRACEFILE_NAME
```

If the output from a debug command produces more than one line of output, the result is sent to a trace file, and a message indicating that the command has completed is relayed to SQL*Plus. If the output from a debug command produces only one line of output, the output is relayed directly to SQL*Plus.

> **Note:** There is currently a limitation when using ORADEBUG that can cause SQL*Plus to hang if you attempt to debug a thread that is blocking on input/output (I/O) until that I/O completes.

8

Authenticating Database Users with Windows

This chapter describes how to authenticate Oracle8*i* database users with Windows.

Specific topics discussed are:

- Windows Native Authentication Overview
- Windows Authentication Protocols
- User Authentication and Role Authorization Methods
- Automatically Enabling Operating System Authentication During Installation
- Administering External Users and Roles
- Administering Enterprise Users and Roles

Windows Native Authentication Overview

The Oracle8*i* database can use Windows user login credentials to authenticate database users. The benefits include:

- Enabling users to connect to multiple Oracle8*i* databases without supplying a user name or password

- Centralizing Oracle8*i* database user authentication and role authorization information in Windows NT or Windows 2000, which frees Oracle8*i* from storing or managing user passwords or role information

Windows native authentication methods (automatically installed with Net8 Server and Net8 Client) enable database user authentication through Windows NT or Windows 2000. This enables client computers to make secure connections to an Oracle8*i* database on a Windows NT or Windows 2000 server. The server then permits the user to perform the database actions on the server.

> **Note:** This chapter describes using Windows native authentication methods with Windows NT 4.0 and Windows 2000. For information on the Secure Socket Layer (SSL) protocol and Oracle Internet Directory, see the *Oracle Advanced Security Administrator's Guide* and *Oracle Internet Directory Administrator's Guide.*

Windows Authentication Protocols

Windows native authentication methods work with Windows authentication protocols to enable access to your Oracle8*i* database. Kerberos is the default authentication protocol for Windows 2000. With Windows NT 4.0, NT LAN Manager (NTLM) is the default protocol.

Client computers do not need to specify an authentication protocol when attempting a connection to an Oracle8*i* database. Instead, the Oracle8*i* database determines the protocol to use, completely transparent to the user. The only Oracle requirement for the client is to ensure that SQLNET.AUTHENTICATION_ SERVICES is set to NTS in the *ORACLE_BASE\ORACLE_ HOME*\NETWORK\ADMIN\SQLNET.ORA file on both the client and database server (this is the default setting for both after installation). For Oracle7 Server and Oracle8 8.0 releases, you must manually set this value using Net8 Assistant.

Your Oracle8*i* database network likely includes client computers and database servers running different Windows operating system and Oracle software releases

in different domains. For example, you may be running an Oracle 8.0.5 client installed on Windows 95 that connects to an Oracle 8.1.6 database installed on a Windows NT 4.0 computer that runs in a Windows 2000 domain. This combination of different releases means that the authentication protocol being used can vary.

This table lists the Oracle software and Windows operating system releases required to enable Kerberos to be the default authentication protocol used:

For The...	This Windows Software is Required...	This Oracle Software is Required...
Client Computer	▪ Windows 2000	▪ Oracle8*i* Client release 8.1.6
Database Computer	▪ Windows 2000	▪ Oracle8*i* database release 8.1.6
Domain	▪ Windows 2000	▪ None

For *all* other combinations of Windows operating system and Oracle software releases used in your network, the authentication protocol used is NTLM.

> **See Also:** Your Windows documentation for more information on each authentication protocol

User Authentication and Role Authorization Methods

This section describes how user login credentials are authenticated and database roles are authorized in Windows NT 4.0 or Windows 2000 domains. User authentication and role authorization are defined as follows:

Feature	Description	More Information
user authentication	The process by which the database uses the user's Windows login credentials to authenticate the user.	*Oracle8i Administrator's Guide*
role authorization	The process of granting an assigned set of roles to authenticated users.	*Oracle8i Administrator's Guide*

In releases prior to 8.1.6, Oracle supported user authentication and role authorization in Windows NT 4.0 domains. This table provides descriptions of these basic features:

Feature	Description	More Information
authentication of external users	Users are authenticated by the database using the user's Windows login credentials that enable them to access the Oracle database without being prompted for additional login credentials.	"External Users and Roles" on page 8-6 "Administering External Users and Roles" on page 8-14
authorization of external roles	Roles are authorized using Windows NT local groups. Once an external role is created, you can grant or revoke that role to a database user. You must set the INIT.ORA parameter OS_ROLES to TRUE to authorize external roles.	

With release 8.1.6, enhancements have been made to support global user authentication and global role authorization with Windows native authentication in Windows 2000 domains using Active Directory. You map multiple enterprise users in a directory server to this global user. These enhancements are available to you *only* if you:

- configure Oracle8*i* to work with a lightweight directory access protocol (LDAP)-compliant directory server such as Active Directory

- are running your Oracle8*i* client and Oracle8*i* database in a Windows 2000 domain

Enterprise user authentication is enabled by setting the OSAUTH_X509_NAME registry parameter to TRUE on the computer on which the Oracle8*i* database is running in a Windows 2000 domain. If this parameter is set to FALSE (the default

setting) in a Windows 2000 domain, then the Oracle8*i* database authenticates the user as an external user (described in "External Users and Roles" on page 8-6). Setting this parameter to TRUE in a Windows NT 4.0 domain does *not* enable you to use enterprise users.

See "Enterprise User Authentication" on page 8-8 for more information on using the OSAUTH_X509_NAME registry parameter.

Which Authentication and Authorization Methods Should I Use?

The user authentication and role authorization methods to use depend upon your Oracle8*i* database environment:

Use...	When...
Enterprise users and roles	You have many users connecting to multiple databases.
	Enterprise users enable you to limit the number of individual database users that you need to create and manage. If you do not create enterprise users, you must create each user individually as a global user in each database to which they require access. Enterprise users require the use of a directory server.
	Use enterprise roles in environments where enterprise users assigned to these roles are located in many geographic regions and must access multiple databases. Enterprise roles enable you to limit the number of individual database roles that you need to create and manage. If you do not create enterprise roles, you must create each role individually in each database. Enterprise roles require the use of a directory server.
External users and roles	You have a smaller number of users accessing a limited number of databases. External users must be created individually in each database, and do not require the use of a directory server.
	External roles must also be created individually in each database, and do not require the use of a directory server. External roles are authorized using group membership of the users in the local groups on the system.

External Users and Roles

The following sections describe external user authentication and external role authorization methods:

- External User Authentication
- External Role Authorization

External User Authentication

This table describes external user authentication features:

Feature	Description
User authentication	Authentication of external users is supported. External users are local users or domain users. External users must be created *individually* in each Oracle8*i* database that they need to access. External users can access the Oracle8*i* database without providing a user name or password. Use external users in environments where users do not need to access multiple databases.
User authentication process	The Oracle8*i* database receives the client user credentials from the Windows authentication protocol and queries the data dictionary to determine if this is a valid user. If the user name is found, the Oracle8*i* database authenticates the user as an external user (with authorized external roles) and permits access.
User administration tools	Use one of the following tools to authenticate external user names: - Oracle Administration Assistant for Windows NT (See "Using Oracle Administration Assistant for Windows NT" on page 8-14.) - Oracle command line tools and the Windows NT User Manager (See "Manually Administering External Users and Roles" on page 8-39.)

Starting with Oracle release 8.1.5, external users created in the database were automatically prefixed with the domain name. For example, for a Windows NT user DOMAIN1\NTUSER1, the Oracle user created in the database is DOMAIN1\NTUSER1. If you want to create the Oracle user in the database without prefixing with the domain name, you first need to set the registry value OSAUTH_PREFIX_DOMAIN in HKEY_LOCAL_ MACHINE\SOFTWARE\ORACLE\HOME*ID* to FALSE.

Note that it is less secure if you do not prefix the domain name. Therefore, if you migrate a database with external users to Oracle8*i* for Windows NT, you have two options:

- Set the registry value OSAUTH_PREFIX_DOMAIN in HKEY_LOCAL_ MACHINE\SOFTWARE\ORACLE\HOME*ID* to FALSE.

- Delete the existing external users and create the new operating system authenticated users in the database using Oracle Administration Assistant for Windows NT.

> **Note:** Oracle Corporation recommends that you do *not* set OSAUTH_PREFIX_DOMAIN to FALSE. Oracle Administration Assistant for Windows NT automatically creates users with their domain prefixed to the user name. If you then attempt to log in with this parameter set to FALSE, you will be unable to do so.

External Role Authorization

This table describes external role authorization features:

Feature	Description
Role authorization	Authorization of external roles is supported. External roles are roles authorized on a single Oracle8*i* database.
	Use external roles in environments where users assigned to these roles do not need to access multiple databases.
User permissions	The permissions (external roles) authorized to a user are those assigned to the local group of which they are a member.
User groups	Users can belong to Windows NT local and domain groups. These groups can be assigned to external roles.

Enterprise Users and Roles

The following sections describe enterprise user authentication and enterprise role authorization methods:

- Enterprise User Authentication
- Enterprise Role Authorization
- Oracle8i Integration with a Directory Server

Enterprise User Authentication

This table describes enterprise user authentication features:

Feature	Description
User authentication	Enterprise users are users created in a directory server (for example, Active Directory) who require access to multiple databases in an enterprise. To access these databases, enterprise users need to be defined in *each* database as a global user.
	For example, assume there is an enterprise user (cn=joe,cn=users,dc=acme,dc=com) who needs access to two databases: Sales and Marketing. This enterprise user must be defined in both databases as a global user.
	Most users do not need their own accounts in the database; they typically need to access only application schemas in a database. This is especially critical in an Internet environment, where a number of users access the same application and there is no need to create global users in the database for all enterprise users.
	In 8.1.6, you can create just one global user in the database and map multiple enterprise users in a directory server to this one global user with Oracle Enterprise Security Manager. See the *Oracle Advanced Security Administrator's Guide* for more information.
User authentication process	Enterprise user authentication is enabled, if you: - Set the OSAUTH_X509_NAME registry parameter to TRUE (See "Oracle8i Integration with a Directory Server" on page 8-10 for instructions.) - Operate your Oracle8*i* database in a Windows 2000 domain - Use Oracle Enterprise Security Manager to map a global user to enterprise users

Feature	Description
User administration tools	Oracle Enterprise Security Manager (an integrated application included in Oracle Enterprise Manager) enables you to create enterprise users, roles, and domains. See the *Oracle Advanced Security Administrator's Guide* for more information on using this tool.
Authentication protocol	Kerberos (if the Windows and Oracle releases match those listed in the table in "Windows Authentication Protocols" on page 8-3; otherwise, NTLM is used).

Enterprise Role Authorization

This table describes enterprise role authentication features:

> **Note:** Enterprise roles are authorized by the directory server, and not by setting the OS_ROLES initialization file parameter to TRUE (the method for enabling external role authorization).

Feature	Description
Role authorization	Authorization of enterprise roles is supported with Oracle8*i* release 8.1.6. An enterprise role is a single role created in a directory server with Oracle Enterprise Security Manager. Use Oracle Enterprise Security Manager to assign multiple global roles and Windows 2000 global and universal users and groups located on multiple databases to this enterprise role. A global role is a role that must be created individually in each Oracle8*i* database.
	For example, an enterprise user can be granted the enterprise role "HR", which contains the global role "HR user" in the human resources database, and the global role "employee" in the corporate information database. If a user changes jobs, his enterprise role assignment is only changed in one place, altering his privileges in multiple databases throughout the enterprise. Also, an administrator can add capabilities to enterprise roles (granted to multiple users) without having to update the authorizations of each individual user.
	Use enterprise roles in environments where users assigned to these roles are located in many geographic regions and must access multiple databases.
	See the *Oracle Advanced Security Administrator's Guide* for more information on creating and storing enterprise roles in a directory server with Oracle Enterprise Security Manager.
User permissions	The permissions authorized to a user are those assigned to the database of which they are a member.
User groups	Users can belong to Windows 2000 global and universal groups. These groups can be assigned to enterprise roles.

Oracle8*i* Integration with a Directory Server

Perform the following tasks to integrate Oracle components with Active Directory. This enables you to take advantage of the new user authentication and role authorization described in "Enterprise Users and Roles" on page 8-8. Note that these enhancements are only available if you are running in a Windows 2000 domain.

- Step 1: Install and Configure Components

- Step 2: Set the OSAUTH_X509_NAME Registry Parameter

- Step 3: Start and Use Oracle Enterprise Security Manager

Step 1: Install and Configure Components

Read Chapter 4, "Using Oracle8i Directory Server Features with Active Directory" and the *Oracle8i Installation Guide for Windows NT* for information on pre-installation, installation, and configuration issues.

Step 2: Set the OSAUTH_X509_NAME Registry Parameter

Set the OSAUTH_X509_NAME registry parameter to enable client users to access the Oracle8*i* database as X.509-compliant enterprise users. This parameter is required *only* if you want to use enterprise users and roles.

Set This Parameter On...	Description
An Oracle8*i* database computer running in a Windows 2000 domain	When set to TRUE, enables a client user name to be identified as an X.509-compliant enterprise user name when connecting to an Oracle8*i* database through Active Directory. User authentication is permitted through the Windows authentication protocol and Active Directory.
	If set to FALSE (the default setting), the client user is identified as an external user and database authorization is permitted through the Windows authentication protocol and Oracle8*i* database data dictionary. No access to Active Directory is available.

To set the OSAUTH_X509_NAME registry parameter:

1. Go to the computer on which the Oracle8*i* database is installed.

2. Choose Start > Run.

3. Enter REGEDT32 in the Open field, and click OK.

   The *Registry Editor* window appears.

4. Go to HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID*.

   where *ID* is the Oracle home that you want to edit.

5. Double-click OSAUTH_X509_NAME.

   A *String Editor* dialog box appears.

6. Set the value to TRUE in the *String* field.

7. Click OK.

8. Choose Exit from the Registry menu.

   The registry editor exits.

Step 3: Start and Use Oracle Enterprise Security Manager

Use Oracle Enterprise Security Manager to create and manage enterprise users, roles, and domains, and assign enterprise users and groups to enterprise roles.

Oracle Enterprise Security Manager is included as an integrated application with Oracle Enterprise Manager. See the *Oracle Advanced Security Administrator's Guide* for information on using this tool. The procedures below describe Windows-unique features for using Oracle Enterprise Security Manager in a Windows 2000 domain.

To use Oracle Enterprise Security Manager:

1. Choose Start > Programs > Oracle - *HOME NAME* > Extended Database Administration > Enterprise Security Manager.

2. Log in when prompted.

3. Use the online help and instructions in *Oracle Advanced Security Administrator's Guide* to use this tool.

4. Review the following issues for using Active Directory.

- The administrator using Oracle Enterprise Security Manager must be a member of the security group OracleDBSecurityAdmin. By default, the administrator who created the Oracle Context (that is, configured the Oracle8*i* database to work with a directory server) is a member of this security group. Only members of this security group are authorized to use all features of Oracle Enterprise Security Manager. To manually add additional users, see "Managing Access Control Lists for Oracle Directory Objects" on page 4-20 for information.

- Select Login from the Directory Server main menu to access a dialog box for selecting the authentication protocol appropriate to your environment:

Select...	If...
NT Native Authentication	Running an Oracle8*i* database on a Windows NT 4.0 or Windows 2000 computer in a Windows 2000 domain with Active Directory.
	This enables Kerberos to be the authentication protocol. Oracle Enterprise Security Manager automatically uses Windows native authentication if running in a Windows 2000 domain.
Simple Authentication	The other available selections do not work. Simple authentication can be used with either Oracle Internet Directory or Active Directory, but is also less secure.

- Windows 2000 groups can appear under the list of Authorized Users in the Navigator window.

Automatically Enabling Operating System Authentication During Installation

When you install your Oracle8*i* database, your Windows user name is automatically added to a Windows NT local group called ORA_DBA. The ORA_DBA local group is:

- Automatically created when your Oracle8*i* database is installed.

- A special Windows NT local group whose members automatically receive the SYSDBA privilege.

This enables you to:

- Connect to any local Oracle8*i* databases without a password by issuing commands such as the following:

    - `CONNECT INTERNAL`

    - `CONNECT / AS SYSDBA`

- Connect to remote Oracle8*i* databases without a password by issuing a command such as the following:

    - `CONNECT /@NET_SERVICE_NAME AS SYSDBA`

        where *NET_SERVICE_NAME* is the net service name of the Oracle8*i* database to which to connect.

- Perform local or remote database administration procedures such as starting and shutting down local databases

- Add additional Windows NT users to ORA_DBA, enabling them to have the SYSDBA privilege, provided you have Administrator privileges

Administering External Users and Roles

There are two methods for administering external users and roles:

- Using Oracle Administration Assistant for Windows NT

- Manually Administering External Users and Roles

> **Note:** Both methods can also administer external users and roles in Windows 2000 domains, but cannot be used to administer enterprise users and roles. See "Administering Enterprise Users and Roles" on page 8-59 for more information on tools available for administering enterprise users and roles.

Using Oracle Administration Assistant for Windows NT

Oracle Administration Assistant for Windows NT is a GUI tool that runs from the Microsoft Management Console. Oracle Administration Assistant for Windows NT enables you to configure the following Oracle database users and roles to be authenticated by the Windows operating system:

- Configure regular Windows NT domain users and global groups as external users to access the Oracle database without a password.

- Configure Windows NT database administrators (with the SYSDBA privilege) to access the Oracle database without a password.

- Configure Windows NT database operators (with the SYSOPER privilege) to access the Oracle database without a password.

- Create and grant local and external database roles to Windows NT domain users and global groups.

Oracle Administration Assistant for Windows NT eliminates the need to manually:

- Create NT local groups that match the database system identifier (SID) and role.

- Assign NT domain users to these local groups.

- Authenticate users in SQL*Plus or Server Manager line mode with the CREATE USER *USERNAME* IDENTIFIED EXTERNALLY syntax.

This section describes how to perform the following tasks with Oracle Administration Assistant for Windows NT:

- Adding a Computer and Saving Your Configuration

- Granting Administrator and Operator Privileges for All Databases on a Computer

- Connecting to a Database

- Viewing Database Authentication Parameter Settings

- Creating a Nonprivileged Database User (External User)

- Creating a Local Database Role

- Creating an External Role

- Granting Administrator and Operator Privileges for a Single Database

> **Note:** Oracle Administration Assistant for Windows NT runs from the Microsoft Management Console, which is automatically included in Windows 2000. If you are using Windows NT 4.0, you must either:
>
> - Install the Microsoft Windows NT 4.0 Option Pack, which includes the Microsoft Management Console
> - Download the Microsoft Management Console from the Microsoft Web site:
>
>   `www.microsoft.com`

> **Note:** If you want to use Oracle Administration Assistant for Windows NT to manage a remote computer, you must have administrator privileges for the remote computer. Oracle Administration Assistant for Windows NT always creates users in the database with the domain name as the prefix. Therefore, if you are managing Oracle 7.*x* or Oracle 8.0.*x* databases remotely, you must set the registry value OSAUTH_PREFIX_DOMAIN in HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID* to TRUE on the remote computer.

Adding a Computer and Saving Your Configuration

When you use Oracle Administration Assistant for Windows NT for the first time, it adds the local computer in the navigation tree. You can then add other computers.

To add a computer to the Microsoft Management Console tree:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Administration Assistant for Windows NT.

   The Microsoft Management Console starts.

2. Double-click Oracle Managed Objects.

   The Computer icon appears.

3. Right-click Computers.

4. Choose New > Computer.

   The *Add Computer* dialog box appears.
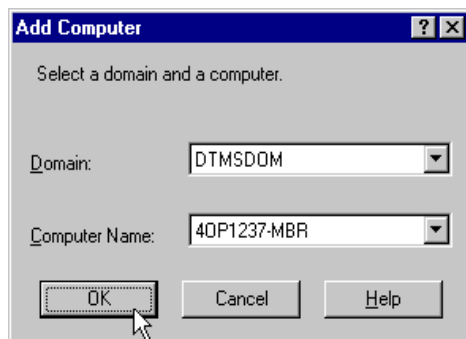
5. Specify the domain and hostname of the computer on which your Oracle database is installed.

6. Click OK.

7. Double-click Computers to display the computer you added.

8. Double-click the computer you added. Several nodes for authenticating database administrators and operators appear:

This Node...	Enables You To...
OS Database Administrators - Computer	Create an operating system-authenticated database administrator (with SYSDBA privileges) for *all* database instances on the computer.
OS Database Operators - Computer	Create an operating system-authenticated database operator (with SYSOPER privileges) for *all* database instances on the computer.

9. Save your configuration in a console file by clicking Save in the Console main menu.

You can now authenticate database administrators and operators for all instances on the computer.

10. See "Granting Administrator and Operator Privileges for All Databases on a Computer" on page 8-18.

Granting Administrator and Operator Privileges for All Databases on a Computer

You can grant database administrator (SYSDBA) and database operator (SYSOPER) privileges to DBAs for *all* databases on a computer.

To grant privileges for all databases on a computer:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Administration Assistant for Windows NT.

   Oracle Administration Assistant for Windows NT starts.

2. Make an appropriate selection:

If You Want to Grant...	Then...	
Database administrator (SYSDBA) privileges	1.	Right click OS Database Administrators - Computer.
	2.	See section "Granting Administrator Privileges for All Databases on a Computer" on page 8-18
Database operator (SYSOPER) privileges	1.	Right click OS Database Operators - Computer.
	2.	See section "Granting Operator Privileges For All Databases on a Computer" on page 8-20

Granting Administrator Privileges for All Databases on a Computer
To grant administrator (SYSDBA) privileges for all databases on a computer:

1. Choose Add/Remove.

The OS Database Administrators - Computer for *hostname* dialog box appears:
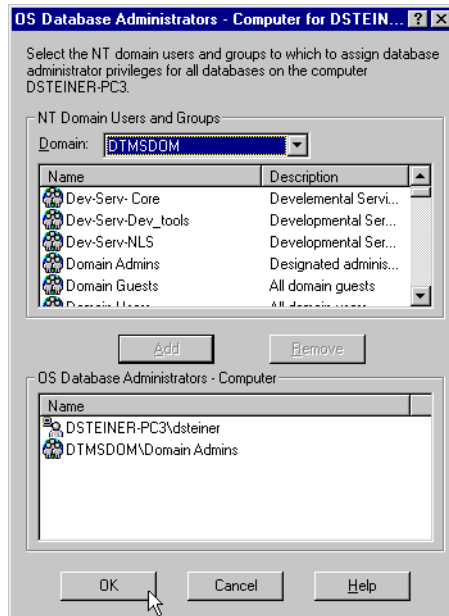
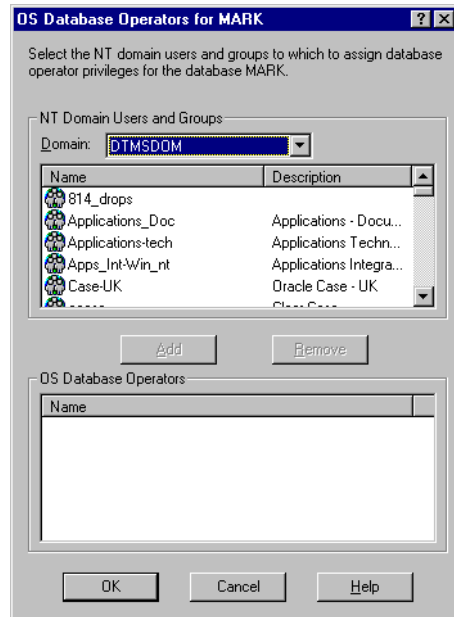**2.** Select the domain of the user to which to grant SYSDBA privileges from the Domain drop-down list box.

3. Select the user.

4. Click Add.

The user now appears in the OS Database Administrators - Computer window.

5. Click OK.

Granting Operator Privileges For All Databases on a Computer
To grant operator (SYSOPER) privileges for all databases on a computer:

1. Choose Add/Remove.

    The OS Database Operators - Computer for *hostname* dialog box appears:
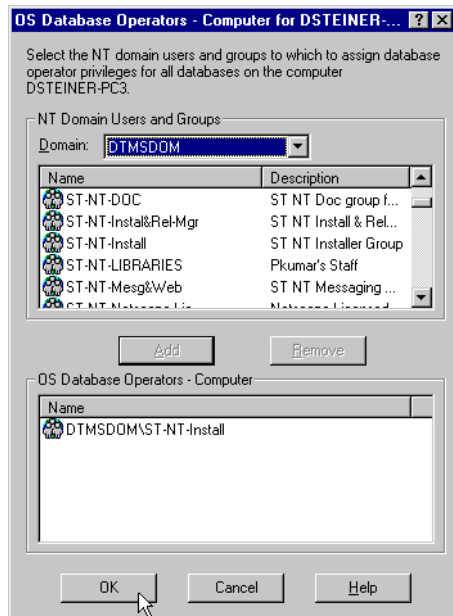
2. Select the domain of the user to which to grant SYSOPER privileges from the Domain drop-down list box.

3. Select the user.

4. Click Add.

    The user now appears in the OS Database Operators - Computer window.

5. Click OK.

Connecting to a Database

Once you connect to a database, you can perform additional authentication tasks:

- Viewing Database Authentication Parameter Settings

- Creating a Nonprivileged Database User (External User) (external user)

- Creating a Local Database Role

- Creating an External Role

- Granting Administrator and Operator Privileges for a Single Database

To connect to a database:

1. Right-click the database instance to access (for example, ORCL) in the Microsoft Management Console scope pane:
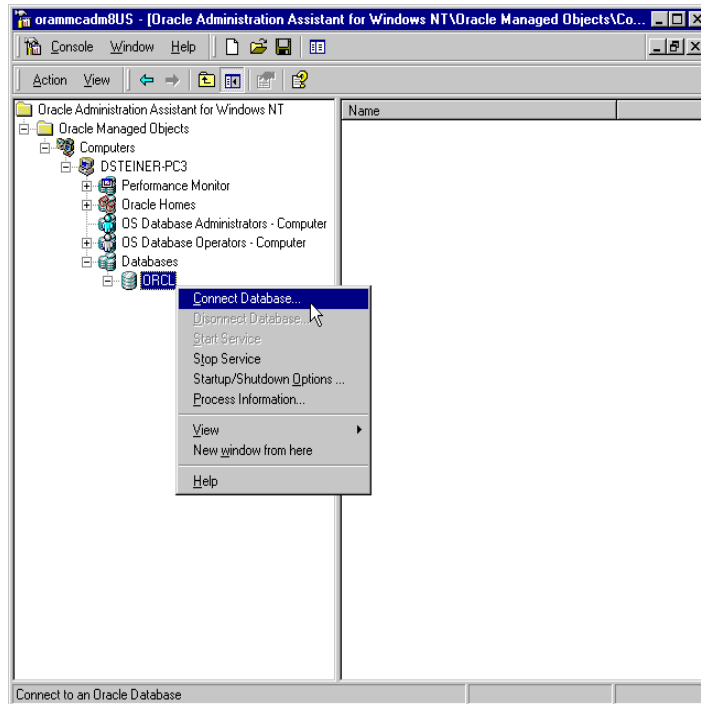
2. Choose Connect Database.

If you connect to the Oracle database, the following Windows NT nodes appear beneath the instance. If these nodes do not appear, double-click the instance.

This Node...	Enables You To...	For More Information...
External OS Users	Authenticate a Windows NT user to access the Oracle database as an external user without being prompted for a password. External users are typically regular database users (non-database administrators) to which you assign standard database roles (such as CONNECT and RESOURCE), but do not want to assign SYSDBA (database administrator) or SYSOPER (database operator) privileges.	See "Creating a Nonprivileged Database User (External User)" on page 8-26
Local Roles	Create a role and have it managed by the database. Once a local role is created, you can grant or revoke that role to a database user.	See "Creating a Local Database Role" on page 8-30
External OS Roles	Create an external role and have it managed by the Windows operating system. Once an external role is created, you can grant or revoke that role to a database user.	See "Creating an External Role" on page 8-32
OS Database Administrators	Authenticate a Windows NT user with SYSDBA privileges for a specific instance on a computer.	See "Granting Administrator and Operator Privileges for a Single Database" on page 8-36
OS Database Operators	Authenticate a Windows NT user with SYSOPER privileges for a specific instance on a computer.	See "Granting Administrator and Operator Privileges for a Single Database" on page 8-36

Troubleshooting Connection Problems

When connecting to a local computer, Oracle Administration Assistant for Windows NT first tries to connect as a SYSDBA to the database using the Bequeath networking protocol. When connecting to a remote computer, Oracle Administration Assistant for Windows NT tries to connect using Windows native authentication as a SYSDBA to the database using the TCP/IP networking protocol (port 1521 and 1526). If it is unsuccessful, the following dialog boxes appear and prompt you to enter information to connect to the database:

This Dialog Box Appears...	Because...
	The Windows NT domain user with which you are attempting to connect to the Oracle database is not recognized as an authenticated user with SYSDBA privileges.
	Enter an Oracle user name and password to access the database.
	To avoid being prompted with this dialog box again, configure your domain user to be a database administrator authenticated by the Windows NT operating system.
	You are not using the TCP/IP networking protocol to connect to a remote Oracle database or the Oracle database is not running. Using SPX or Named Pipes causes this dialog box to appear each time you attempt a remote connection.
	Change to the TCP/IP networking protocol and use default port 1521 or 1526. Otherwise, this dialog appears every time. Ensure also that the Oracle database is started.
	1. Enter the net service name with which to connect to your Oracle database. You must enter a net service name regardless of the authentication method you select.
	2. If you want to access the database with an Oracle user name and password, select the Database Authenticated option. This user name and password must exist in the Oracle database and have the SYSDBA privilege.
	3. If you want to access the database with the Windows NT domain user with which you are currently logged in, select the OS Authenticated Connection as SYSDBA option. This domain user must already be recognized by Windows NT as an authenticated user with SYSDBA privileges. Otherwise, your logon fails.
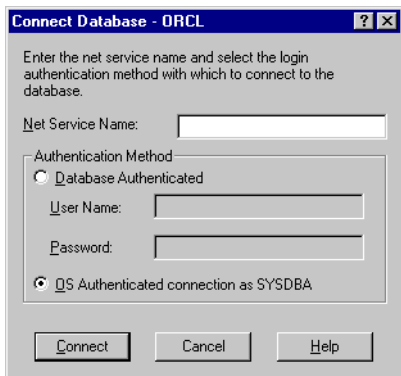
Viewing Database Authentication Parameter Settings

To view database authentication parameter settings:

1. Right click the database.

2. Choose Properties.

3. View the following database properties:

Parameter	Description
OS_AUTHENT_PREFIX	OS_AUTHENT_PREFIX is an INIT.ORA file parameter that authenticates external users attempting to connect to the Oracle database with the user's Windows NT user name and password. The value of this parameter is attached to the beginning of every user's Windows user name. By default, the parameter is set to none ("") during Oracle8*i* database creation. Create Oracle users in the database without the prefix OPS$, which was needed for Oracle7 and Oracle8 8.0.*x*.
	Therefore, a Windows domain user name of FRANK is authenticated as user name FRANK. You can set this parameter to an appropriate value. For example, if you set this parameter to XYZ, the Windows NT domain user FRANK is authenticated as user XYZFRANK.

Parameter	Description
OS_ROLES	OS_ROLES is an INIT.ORA file parameter that, if set to TRUE, enables the Windows NT operating system to manage the authorization of external roles for database users. By default, OS_ROLES is set to FALSE. You must set OS_ROLES to TRUE and restart your Oracle database before you can create external roles. If OS_ROLES is set to FALSE, the Oracle database manages the granting and revoking of roles for database users. See section "Understanding the OS_ROLES Parameter" on page 8-25 for more information.

Understanding the OS_ROLES Parameter

OS_ROLES is a parameter in the INIT.ORA file that, if set to TRUE, enables the Windows NT operating system to manage the authorization of external roles for database users. You must set OS_ROLES to TRUE and restart your Oracle database before you can create external roles.

If OS_ROLES is set to FALSE, the Oracle database manages the granting and revoking of roles for database users.

If OS_ROLES is set to TRUE and you assign an external role to an NT global group, it is granted only at the global group level, and not at the level of the individual user in this global group. This means that you cannot revoke or edit the external role assigned to an individual user in this global group through the Roles tab of the *Domain\User Name Properties* dialog box at a later time. Instead, you must use the Assign External OS Roles to an NT Global Group field in the dialog box to revoke the external role from this global group (and therefore all its individual users).

External roles assigned to an individual domain user or local roles (with OS_ROLES set to FALSE) assigned to an individual domain user or NT global group are not affected by this issue, and can be edited or revoked.

If OS_ROLES is set to TRUE, you can not grant local roles in the database to any database user. You must grant the roles through Windows NT. See "Creating a Local Database Role" on page 8-30 and "Creating an External Role" on page 8-32 for more information.

Creating a Nonprivileged Database User (External User)

You can create a nonprivileged database user (external user).

To create a nonprivileged database user:

1. Follow the steps in "Connecting to a Database" on page 8-21 to connect to a database.

2. Right click External OS Users.

3. Choose Create.
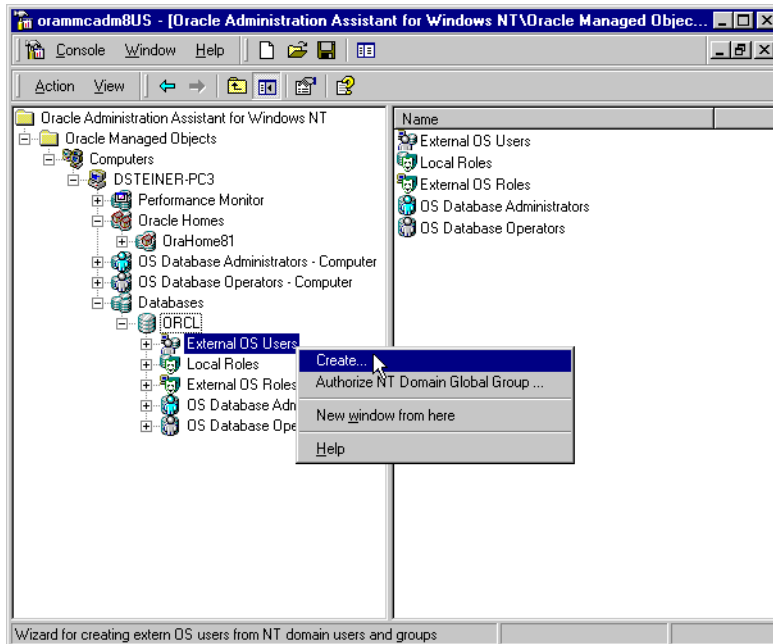
The Create External OS User Wizard starts:
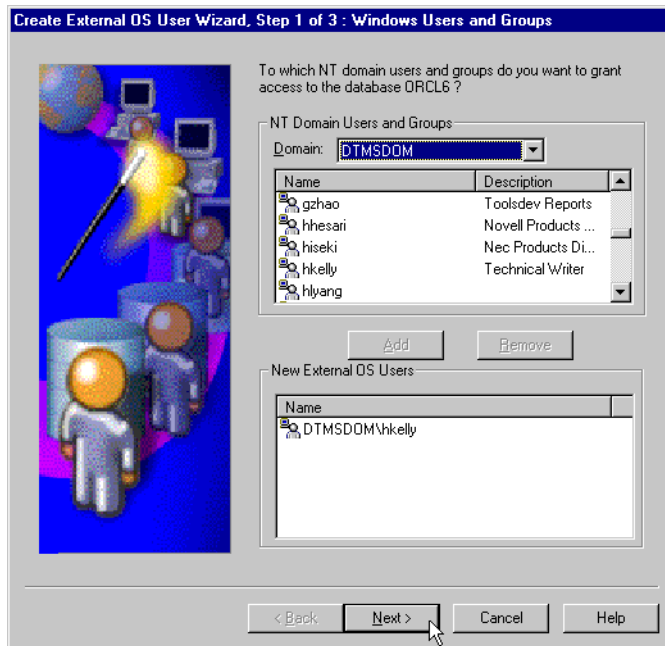
4. Select the domain in which your Windows NT domain users and global groups are located.

5. Select the Windows NT domain users and global groups to which to grant access to the database.

6. Click Add.

   The user now appears in the New External OS Users window.

   > **Note:** If you select an NT global group for authentication when using Oracle Administration Assistant for Windows NT, all users currently in the group are added to the Oracle database. If at a later time, you use a Windows NT tool to add or remove users in this NT global group, these updates are not reflected in the Oracle database. The newly added or removed users must be explicitly added or removed in the Oracle database with Oracle Administration Assistant for Windows NT.

7. Click Next.
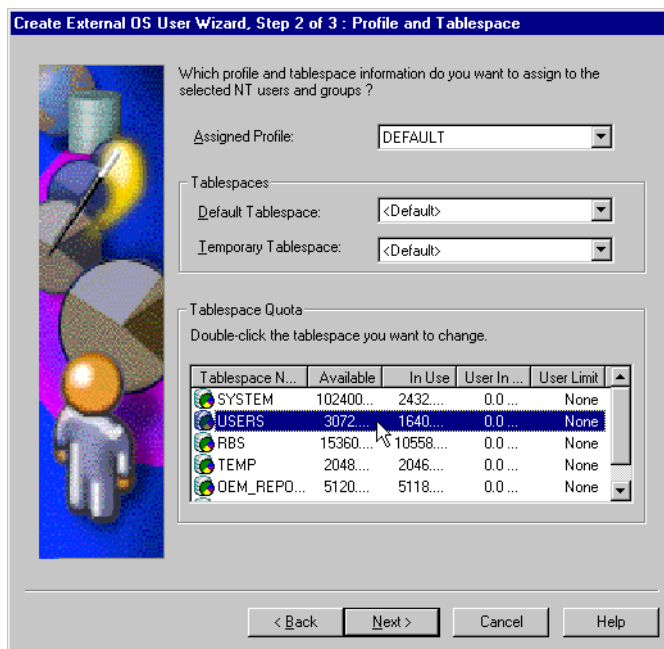
8. Select a profile for the new external users. A profile is a named set of resource limits. If resource limits are enabled, Oracle limits database usage and instance resources to whatever is defined in the user's profile. You can assign a profile to each user, and a default profile to all users who do not have specific profiles.

9. Double-click the tablespace to assign a tablespace quota in the Tablespace Quota window. This assigns profile and tablespace information to the users, and grants database roles.

10. Click Next.
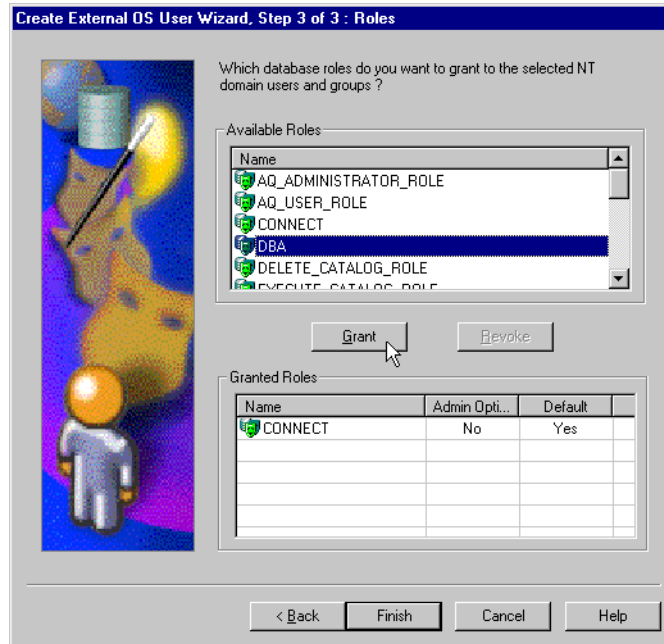
11. Select the database roles to grant to the new external users.

12. Click the Grant button.

13. Click Finish.

14. Right click the external user for which you want to view information and select Properties.

   The assigned properties appear.

Creating a Local Database Role

You can create a local database role.

To create a local database role:

1. Follow the steps in "Connecting to a Database" on page 8-21 to connect to a database.

2. Right click Local Roles for the database for which you want to create a local role.

3. Click Create.

   The *Create Local Role* wizard appears:
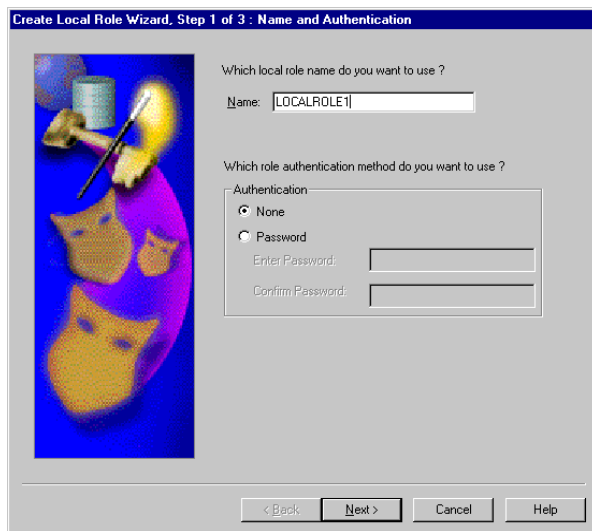
4. Enter a local role name to use. A local role is a role that is managed by the Oracle database.

5. Select None if you want a user to use this local role without being required to enter a password.

6. Select Password if you want the use of this role to be protected by a password. These roles can only be used by supplying an associated password with the SET ROLE command. See the *Oracle8i Administrator's Guide* for additional information.

7. Enter the password to use with this role.

8. Confirm the password by entering it a second time.

9. Click Next.
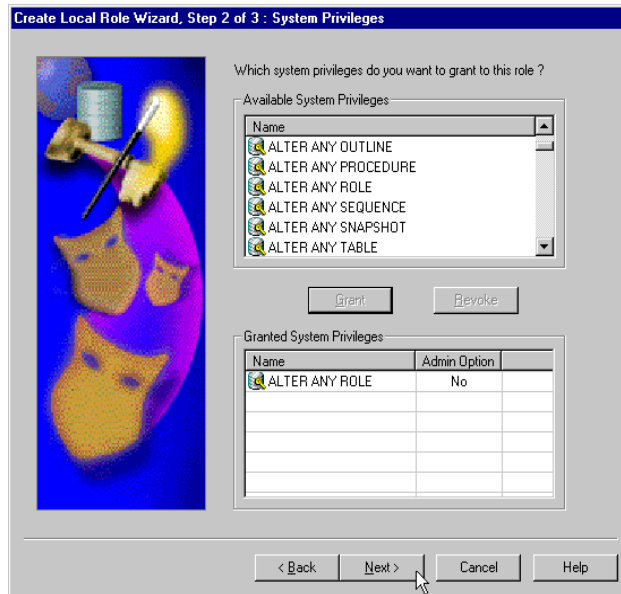
10. Select appropriate system privileges to assign to the local role.

11. Click Grant to grant the selected system privileges to the local role.

    The Granted System Privileges field displays the list of system privileges granted to the local role. To revoke a system privilege, make an appropriate selection, then click Revoke.

12. If you want to grant the Admin Option to this role, click the value in the Admin Option column to display a drop-down list box. This enables you to select Yes.

13. Click Next.

14. Select appropriate roles to assign to the local role. Both local roles and external roles appear in this list.
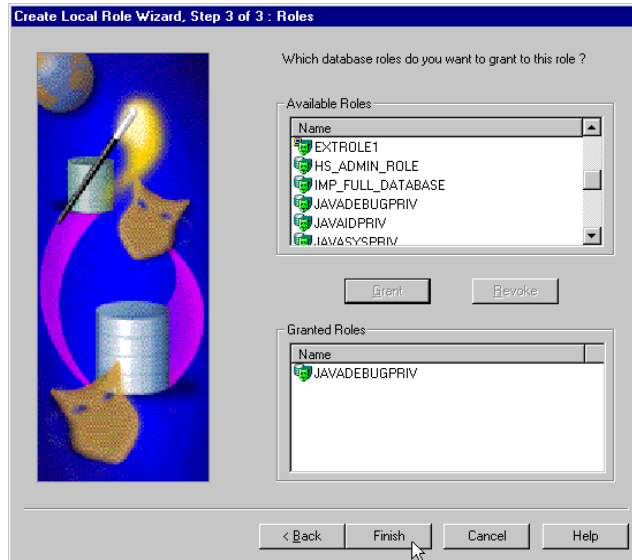
15. Click Grant to grant the selected roles to the role.

   The Granted Roles field displays the list of roles granted to the role. Both local roles and external roles can appear in this list. To revoke roles, make appropriate selections, then click Revoke.

16. Click Finish.

Creating an External Role

You can create external roles.

To create an external role:

1. Follow the steps in "Connecting to a Database" on page 8-21 to connect to a database.

1. Right click External OS Roles for the database for which to create an external role.

2. Click Create.

> **Note:** This wizard is only available if you set the INIT.ORA parameter OS_ROLES to TRUE and restart the Oracle database.
>
> "Authentication: External" appears on this page to indicate that only external roles can be created.
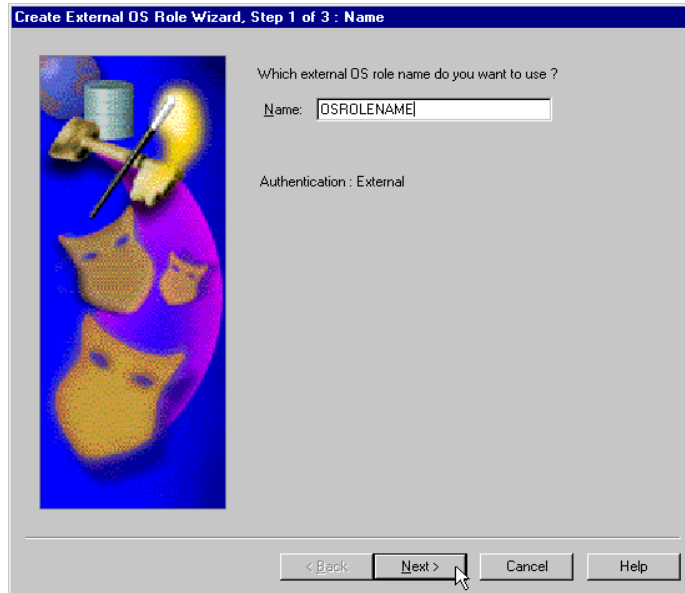
3. Enter an external role name to use. An external role is a role that is managed by the Windows operating system.

4. Click Next.
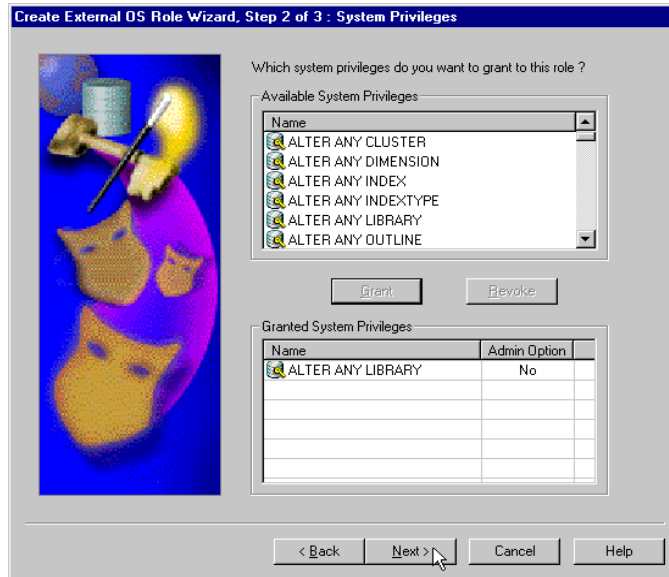
5. Select appropriate system privileges to assign to the external role.

6. Click Grant to grant the selected system privileges to the external role.

7. The Granted System Privileges field displays the list of system privileges granted to the external role. To revoke a system privilege, make an appropriate selection, then click Revoke.

8. If you want to grant the Admin Option to this role, click the value in the Admin Option column to display a drop-down list box. This enables you to select Yes.

9. Click Next.
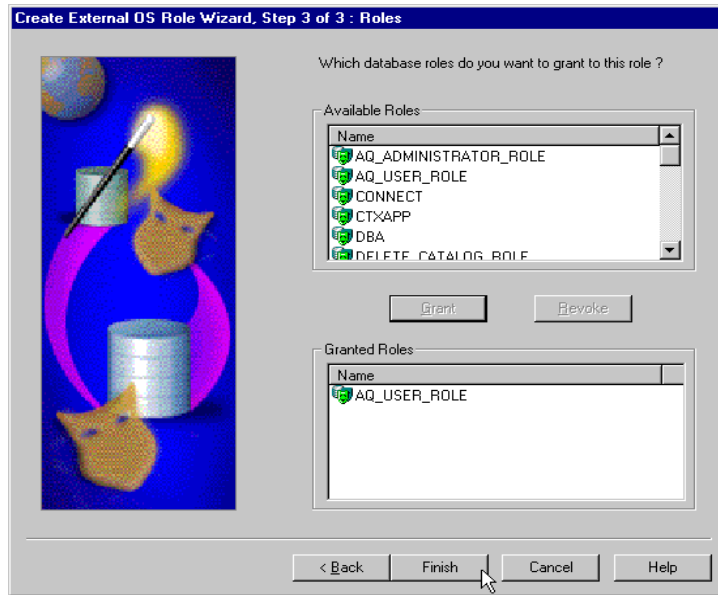
10. Select appropriate roles to assign to the external role.

11. Click Grant to grant the selected roles to the external role. Both local roles and external roles appear in this list.

    The Granted Roles field displays the list of roles granted to the external role.

12. Click Finish.

Granting Administrator and Operator Privileges for a Single Database

You can grant database administrator (SYSDBA) and database operator (SYSOPER) privileges to DBAs for a single database on a computer.

To grant privileges for a single database:

1. Follow the steps in "Connecting to a Database" on page 8-21 to connect to a database.

2. Right-click the database to access (for example, ORCL) in the Microsoft Management Console scope pane.

3. Choose Connect Database.

   Several icons, including OS Database Administrators and OS Database Operators, appear:

4. Make an appropriate selection:

If You Want to Grant...	Then...	
Database administrator (SYSDBA) privileges	1.	Right click OS Database Administrators.
	2.	See section "Granting Administrator Privileges for a Single Database" on page 8-37
Database operator (SYSOPER) privileges	1.	Right click OS Database Operators.
	2.	See section "Granting Operator Privileges for A Single Database" on page 8-38

Granting Administrator Privileges for a Single Database

To grant administrator (SYSDBA) privileges for a single database:

1. Choose Add/Remove.

The OS Database Administrators for *instance* dialog box (MARK in this example) appears:
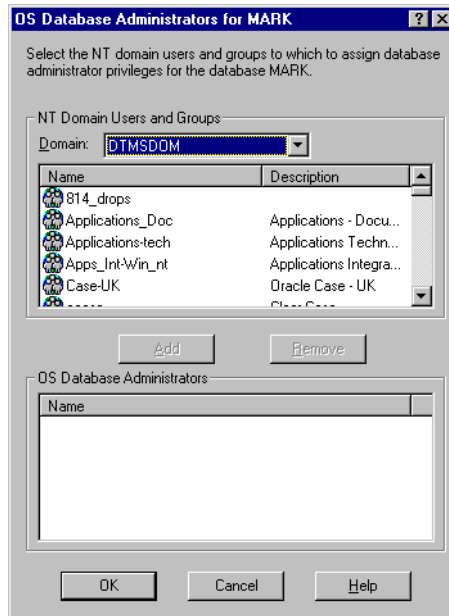
2. Select the domain of the user to which to grant SYSDBA privileges from the NT Domain Users and Groups drop-down list box.

3. Select the user. The user now appears in the OS Database Administrators window.

4. Click OK.

Granting Operator Privileges for A Single Database
To grant operator (SYSOPER) privileges for a single database:

1. Choose Add/Remove.

   The OS Database Operators for *instance* dialog box (MARK in this example) appears:

2. Select the domain of the user to which to grant SYSOPER privileges from the NT Domain Users and Groups drop-down list box.

3. Select the user.

4. Click Add.

   The user now appears in the OS Database Operators window.

5. Click OK.

Manually Administering External Users and Roles

Manual configuration involves using Oracle command line tools, editing the registry with REGEDT32, and creating local groups in Windows NT User Manager. This enables you to:

- Configure nonprivileged Windows NT users (external users) to access the Oracle database without a password.

- Configure Windows NT database administrators (with the SYSDBA privilege) to access the Oracle database without a password.

- Configure Windows NT database operators (with the SYSOPER privilege) to access the Oracle database without a password.

- Create and grant local and external database roles to Windows NT domain users and global groups.

This section describes:

- Creating a Nonprivileged Database User (External User)

- Granting Administrator and Operator Privileges for Databases

- Connecting as INTERNAL Without a Password

- Creating an External Role

> **Note:** Use extreme care when manually configuring administrators, operators, users, and roles to be authenticated by the operating system. If possible, use Oracle Administration Assistant for Windows NT to perform configuration procedures.

Creating a Nonprivileged Database User (External User)

This section describes how to authenticate nonprivileged database users (nondatabase administrators) using Windows NT so that a password is not required when accessing the database. When you use Windows NT to authenticate nonprivileged database users, your database relies solely on Windows NT to restrict access to database user names. In the steps below, the following Windows NT user names are authenticated:

User Name	This User...
Local user FRANK	Logs into their local Windows NT client computer to access an Oracle8*i* database. The database can be on a different computer. To access other databases and resources on other computers, the local user must provide a user name and password each time.
Domain user FRANK on domain SALES	Logs into a domain (SALES in the steps below) that includes many other Windows NT computers and resources, one of which contains an Oracle8*i* database. The domain user can access all the resources the domain provides with a single user name and password.

The local and domain user name FRANK and the domain SALES are used in the steps below. Substitute the appropriate local and domain user name and domain name for your environment.

Follow the steps below to connect without a password as a nonprivileged database user:

- Step 1: Perform Authentication Tasks on the Oracle8i Database Server

- Step 2: Perform Authentication Tasks on the Client Computer

Step 1: Perform Authentication Tasks on the Oracle8*i* Database Server

To perform authentication tasks on an Oracle8*i* database server:

1. Add the OS_AUTHENT_PREFIX parameter to your INIT.ORA file.

   The OS_AUTHENT_PREFIX value is prefixed to local or domain user names attempting to connect to the server with the user's operating system name and password. The prefixed user name is compared with the Oracle user names in the database when a connection request is attempted. Using the OS_AUTHENT_PREFIX parameter with Windows native authentication methods is the recommended method for performing secure, trusted client connections to your server.

2. Set OS_AUTHENT_PREFIX to an appropriate value. Values are case insensitive. For example:

Set OS_AUTHENT_PREFIX to...	Result
XYZ	XYZ is prefixed to the beginning of the Windows NT user name (for example, XYZFRANK for local user FRANK or XYZSALES\FRANK for domain user FRANK on domain SALES).
	Note: XYZ is only an example of an acceptable parameter value. Use a value appropriate to your environment.
""	This is recommended, as it eliminates the need for any prefix to the Windows NT user names (for example, FRANK for local user FRANK or SALES\FRANK for domain user FRANK on domain SALES).
Not included in INIT.ORA file	The value defaults to OPS$ (for example, OPS$FRANK for local user FRANK or OPS$SALES\FRANK for domain user FRANK on domain SALES).

The parameter value XYZ is used in the steps below. Substitute XYZ with the value you set for OS_AUTHENT_PREFIX.

3. Use User Manager to create a Windows NT local or domain user name for FRANK (if the appropriate name does not currently exist). See your Windows NT documentation or your network administrator if you do not know how to do this.

4. Follow the substeps below to create a new registry parameter *only* if you are *not* authenticating a domain name with a user (for example, just FRANK instead of FRANK on domain SALES). Otherwise, go to step 5.

a. Start the registry editor from the MS-DOS command prompt:

```
C:\> REGEDT32
```

b. Go to HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\ HOME*ID,* where *ID* is the Oracle home directory you want to edit.

c. Choose the Add Value option in the Edit menu.

The *Add Value* dialog box appears:
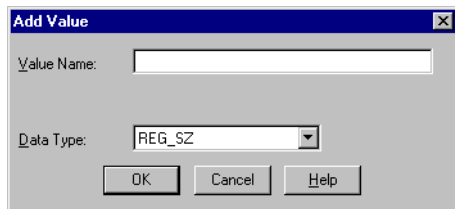
   d. Enter OSAUTH_PREFIX_DOMAIN in the Value Name field.

   e. Choose REG_EXPAND_SZ from the Data Type drop-down list box.

   f. Click OK.

   The *String Editor* dialog box appears:
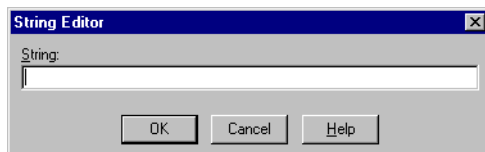
   g. Enter TRUE in the String field to enable authentication at the domain level.

   TRUE enables the server to differentiate between multiple FRANK user names, whether they are local user FRANK, domain user FRANK on SALES, or domain user FRANK on another domain in your network. Entering FALSE causes the domain to be ignored and local user FRANK to become the default value of the operating system user returned to the server.

   h. Click OK.

   The Registry Editor adds the parameter.

   i. Choose Exit from the registry menu.

   The registry exits.

5. Ensure that you have the following line in your *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN\SQLNET.ORA file:

```
SQLNET.AUTHENTICATION_SERVICES = (NTS)
```

6. Start SQL*Plus:

```
C:\> SQLPLUS
```

7. Connect to the database with the SYSTEM database administrator (DBA) name:

```
SQL> CONNECT

Enter user-name: SYSTEM/PASSWORD
```

Unless you have changed it, the SYSTEM password is MANAGER by default.

8. Create an operating system-authenticated user by entering the following:

If Authenticating a...	Then Enter...
Local user name	`SQL> CREATE USER XYZFRANK IDENTIFIED EXTERNALLY;`
Domain user name	`SQL> CREATE USER "XYZSALES\FRANK" IDENTIFIED EXTERNALLY;`

Where:	Is the...
XYZ	Value set for the OS_AUTHENT_PREFIX initialization parameter.
FRANK	Windows NT local user name.
SALES\FRANK	Domain name and Windows NT domain user name. The double quotes are required and the entire syntax *must* be in uppercase.

9. Grant the Windows NT local user FRANK or domain user FRANK appropriate database roles:

If Authenticating a...	Then Enter...
Local user name	`SQL> GRANT RESOURCE TO XYZFRANK;`
	`SQL> GRANT CONNECT TO XYZFRANK;`
Domain user name[1]	`SQL> GRANT RESOURCE TO "XYZSALES\FRANK";`
	`SQL> GRANT CONNECT TO "XYZSALES\FRANK";`

[1] Enter the syntax for domain users in uppercase and with double quotes around the domain user name.

10. Connect to the database with the INTERNAL DBA name:

```
SQL> CONNECT INTERNAL
```

11. Shut down the database:

```
SQL> SHUTDOWN
```

12. Restart the database:

```
SQL> STARTUP
```

This causes the change to the OS_AUTHENT_PREFIX parameter value to take affect.

Step 2: Perform Authentication Tasks on the Client Computer

To perform authentication tasks on the client computer:

1. Create Windows NT local or domain user name FRANK with the same user name and password that exist on the Windows NT server (if the appropriate name does not currently exist).

2. Ensure that you have the following line in your *ORACLE_BASE\ ORACLE_HOME*\NETWORK\ADMIN\SQLNET.ORA file:

```
SQLNET.AUTHENTICATION_SERVICES = (NTS)
```

3. Use Net8 Assistant to configure a network connection from your client computer to the Windows NT server on which your Oracle8*i* database is installed. See the *Net8 Administrator's Guide* for instructions.

4. Start SQL*Plus:

```
C:\> SQLPLUS
```

5. Connect to your Windows NT server:

```
SQL> CONNECT /@NET_SERVICE_NAME
```

where *NET_SERVICE_NAME* is the Net8 network service name for the Oracle8*i* database that you created in Step 3.

The Oracle8*i* database searches the data dictionary for an automatic login user name corresponding to the Windows NT local or domain user name, verifies it, and allows you to connect as XYZFRANK or XYZSALES\FRANK.

6. Verify that you have connected to the Oracle8*i* database as local or domain user FRANK by viewing the roles assigned in Step 9 of "Step 1: Perform Authentication Tasks on the Oracle8i Database Server" on page 8-43.

```
SQL> SELECT * FROM USER_ROLE_PRIVS;
```

which outputs for local user FRANK:

```
USERNAME                       GRANTED_ROLE                   ADM DEF OS_

------------------------------ ------------------------------ --- --- ---

XYZFRANK                       CONNECT                        NO  YES NO

XYZFRANK                       RESOURCE                       NO  YES NO

2 rows selected.
```

or, for domain user FRANK:

```
USERNAME                       GRANTED_ROLE                   ADM DEF OS_

------------------------------ ------------------------------ --- --- ---

XYZSALES\FRANK                 CONNECT                        NO  YES NO

XYZSALES\FRANK                 RESOURCE                       NO  YES NO

2 rows selected.
```

As the Oracle8*i* user name is the whole name XYZFRANK or XYZSALES\FRANK, all objects created by XYZFRANK or XYZSALES\FRANK (that is, tables, views, indexes, and so on) are prefixed by this name. For another user to reference the table SHARK owned by XYZFRANK, for example, the user must enter:

```
SQL> SELECT * FROM XYZFRANK.SHARK
```

> **Attention:** Automatic authorization is supported for all Net8 protocols.

Granting Administrator and Operator Privileges for Databases

This section describes how to enable Windows NT to grant the database administrator (SYSDBA) and database operator (SYSOPER) privileges to DBAs. This enables DBAs to issue the following commands from a client computer and connect to the Oracle8*i* database without entering a password:

- `CONNECT / AS SYSOPER`

- `CONNECT / AS SYSDBA`

To enable this feature, the Windows NT local or domain user name of the client must belong to one of the following four Windows NT local groups on the server:

Local Group	This Local Group Includes All...
ORA_OPER	SYSOPER database privileges; applicable for all databases on a computer.
ORA_DBA [1]	SYSDBA database privileges; applicable for all databases on a computer.
ORA_*SID*_DBA	SYSDBA database privileges; applicable only for a single database on a computer (identified by the SID).
ORA_*SID*_OPER	SYSOPER database privileges; applicable only for a single database on a computer (identified by the SID).

[1] ORA_DBA is automatically created during installation. See section "Automatically Enabling Operating System Authentication During Installation" on page 8-13 for information.

The SYSOPER and SYSDBA privileges are mapped to the following Windows NT local groups:

This Privilege...	Maps to the Local Group...
SYSOPER	ORA_*SID*_OPER, ORA_OPER
SYSDBA	ORA_*SID*_DBA, ORA_DBA, ORA_*SID*_OPER, ORA_OPER

Follow the steps below to connect as SYSOPER or SYSDBA without a password:

- Step 1: Perform Authentication Tasks on the Oracle8i Database Server

- Step 2: Perform Authentication Tasks on the Client Computer

Step 1: Perform Authentication Tasks on the Oracle8*i* Database Server

To perform authentication tasks on the Oracle8*i* database server:

1. Open User Manager on the Windows NT server where your Oracle8*i* database is installed.

2. Choose New Local Group from the User Menu.

   The *New Local Group* dialog box appears.

3. Enter the appropriate Windows NT local group name in the Group Name field. For this example, the SID entered is ORCL.
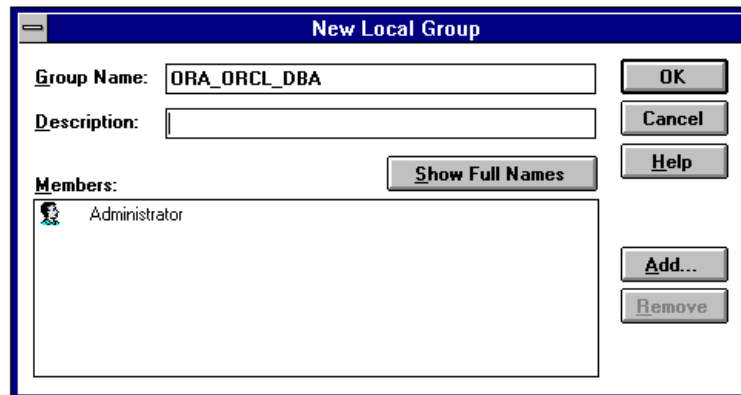
4. Click Add.

The *Add Users and Groups* dialog box appears:
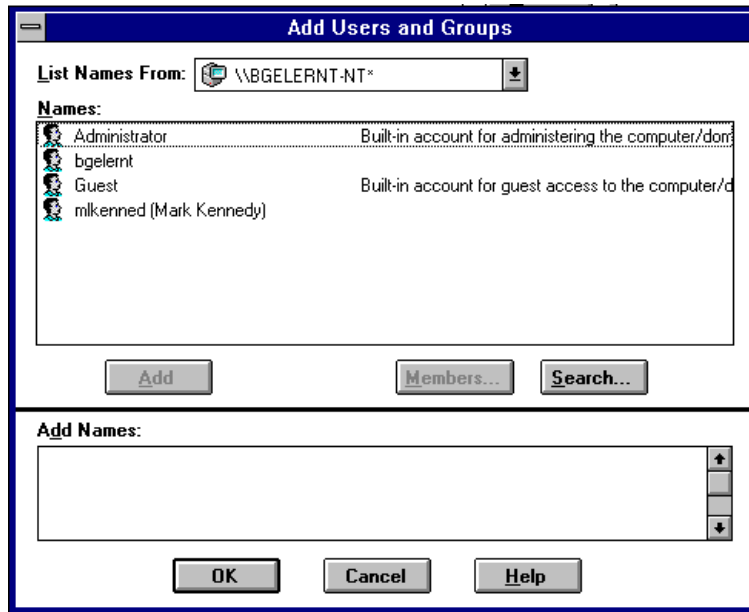
5. Select an appropriate Windows NT user from the Names field and click Add.

6. Click OK.

Your selection is added to the Members field of the *New Local Group* dialog box:
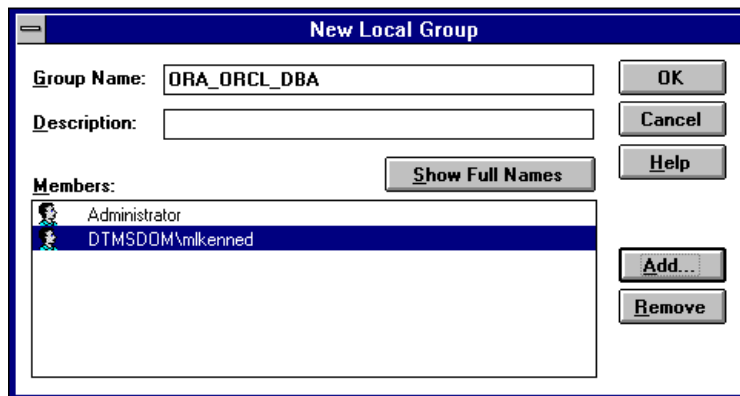
7. Click OK.

8. Exit User Manager.

9. Ensure that you have the following line in your *ORACLE_BASE\ORACLE_ HOME*\NETWORK\ADMIN\SQLNET.ORA file:

   ```
   SQLNET.AUTHENTICATION_SERVICES = (NTS)
   ```

10. In the registry in HKEY_LOCAL_ MACHINE\SOFTWARE\ORACLE\HOME*ID* set the parameter OSAUTH_ PREFIX_DOMAIN to TRUE.

Step 2: Perform Authentication Tasks on the Client Computer

To perform authentication tasks on the client computer:

1. Create a Windows NT local or domain user name with the same user name and password that exist on the Windows NT server (if the appropriate user name does not currently exist).

2. Ensure that you have the following line in your *ORACLE_BASE\ORACLE_ HOME*\NETWORK\ADMIN\SQLNET.ORA file:

   ```
   SQLNET.AUTHENTICATION_SERVICES = (NTS)
   ```

3. Use Net8 Assistant to configure a network connection from your client computer to the Windows NT server on which your Oracle8*i* database is installed. See *Net8 Administrator's Guide* for instructions.

4. Start SQL*Plus:

   ```
   C:\> SQLPLUS
   ```

5. Connect to the Oracle8*i* database:

   ```
   SQL> SET INSTANCE NET_SERVICE_NAME
   ```

   where *NET_SERVICE_NAME* is the Net8 network service name for the Oracle8*i* database that you created in Step 3.

6. Connect as SYSOPER or SYSDBA based on the local group you specified in step 3 of "Step 1: Perform Authentication Tasks on the Oracle8i Database Server":

If The Local Group Is...	Then Enter...

ORA_DBA or ORA_*SID*_DBA	`SQL> CONNECT / AS SYSOPER`
	or
	`SQL> CONNECT / AS SYSDBA`

ORA_OPER or ORA_*SID*_OPER	`SQL> CONNECT / AS SYSOPER`

You are connected to the Windows NT server. If you connect with SYSDBA, you are given DBA privileges.

Connecting as INTERNAL Without a Password

This section describes how to connect as INTERNAL without a password. If you installed your Oracle8*i* database, your Windows NT user name was automatically added to a Windows NT local group called ORA_DBA. This enables you to automatically connect as INTERNAL without a password. However, if you or the user to which to assign this feature did not install your Oracle8*i* database, then you must follow the instructions in this section.

To connect as INTERNAL without a password, you must create one of the following new local Windows NT user groups and add a Windows NT operating system local or domain user to that group.

Local Group	This Local Group Includes All...
ORA_DBA [1]	SYSDBA database privileges. This group is applicable for all SIDs.
ORA_*SID*_DBA	SYSDBA database privileges. This group is applicable only for the SID specified in the name.

[1] ORA_DBA is automatically created during installation. See section "Automatically Enabling Operating System Authentication During Installation" on page 8-13 for information.

This enables you to log into a local computer or a Windows NT domain. In the domain, your Oracle8*i* database is just one of many resources to which you have access. Once you access this domain, you are automatically validated as an authorized DBA who can access the Oracle8*i* database without a password.

Follow the steps below to connect as INTERNAL without a password:

■   Step 1: Perform Authentication Tasks on the Oracle8i Database Server

■   Step 2: Perform Authentication Tasks on the Client Computer

Step 1: Perform Authentication Tasks on the Oracle8*i* Database Server

To perform authentication tasks on the Oracle8*i* database server:

1. Create a Windows NT user name (local or domain) if one does not already exist.

2. Ensure that you have the following line in your *ORACLE_BASE\ORACLE_HOME*NETWORK\ADMIN\SQLNET.ORA file:

   ```
   SQLNET.AUTHENTICATION_SERVICES = (NTS)
   ```

3. Open User Manager.

4. Go to New Local Group from the User Menu.

   The *New Local Group* dialog box appears.

5. Enter the ORA_*SID*_DBA or ORA_DBA Windows NT local group name in the Group Name field. For this example, the SID entered is ORCL:
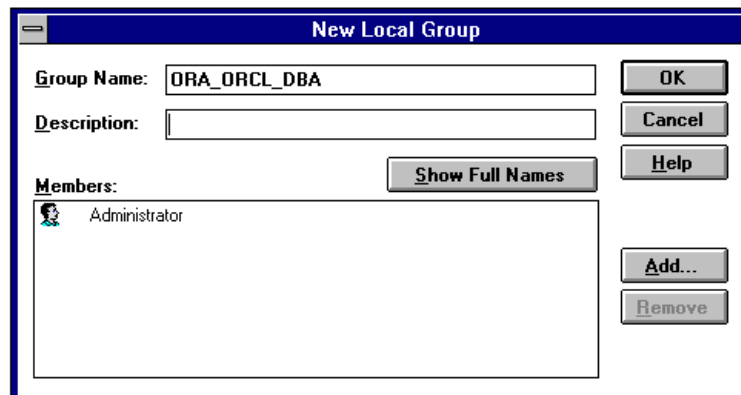
6. Click Add.

The *Add Users and Groups* dialog box appears:
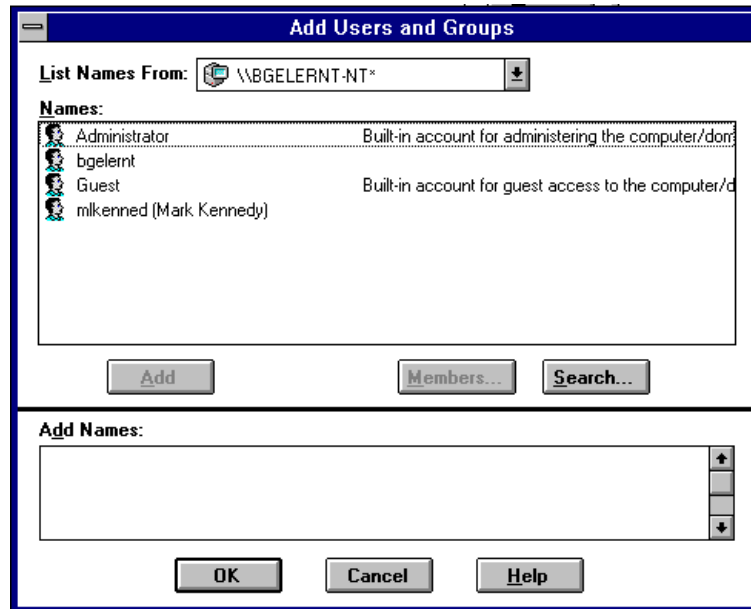
7. Select an appropriate Windows NT local or domain user from the Names field and click Add.

8. Click OK.

   Your selection is added to the Members field of the *New Local Group* dialog box:
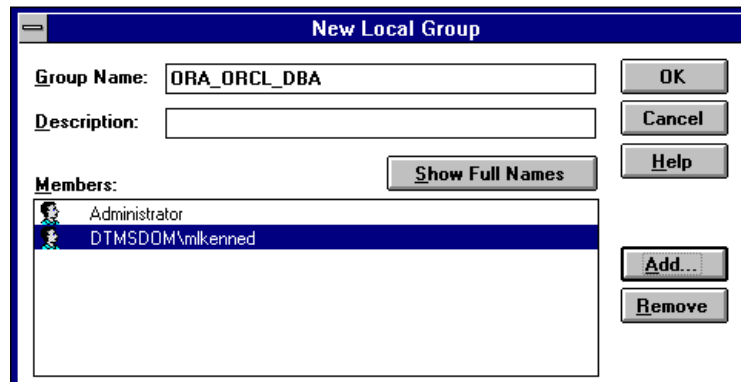
9. Click OK.

10. Exit User Manager.

Step 2: Perform Authentication Tasks on the Client Computer

To perform authentication tasks on the client computer:

1. Create a Windows NT local or domain user name with the same user name and password that exist on the Windows NT server (if the appropriate user name does not currently exist).

2. Ensure that you have the following line in your *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN\SQLNET.ORA file:

   ```
   SQLNET.AUTHENTICATION_SERVICES = (NTS)
   ```

3. Use Net8 Assistant to configure a network connection from your client computer to your Oracle8*i* database. See *Net8 Administrator's Guide* for instructions.

4. Start SQL*Plus:

   ```
   C:\> SQLPLUS
   ```

5. Connect to the Oracle8*i* database:

   ```
   SQL> SET INSTANCE NET_SERVICE_NAME
   ```

   where *NET_SERVICE_NAME* is the Net8 network service name for the Oracle8*i* database that you created in Step 3.

6. Connect to your Windows NT server:

   ```
   SQL> CONNECT INTERNAL
   ```

   You are connected to the Windows NT server.

Creating an External Role

This section describes how to grant Oracle8*i* database roles to users directly through Windows NT (known as external roles). When you use Windows NT to authenticate users, Windows NT local groups can grant these users external roles. Through User Manager, you can create, grant, or revoke external roles to users.

All privileges for these roles are active when the user connects. When using external roles, all roles are granted and managed through the operating system. You cannot use both external roles and Oracle roles at the same time. For example:

If You...	Then...
1. Enable external roles.	You only receive the roles granted to DTMSDOM\FRANK, and *not* the roles granted to SCOTT.
2. Log onto a Windows NT domain with your domain user name; for example, SALES\FRANK, where SALES is the domain name and FRANK is the domain user name.	
3. Connect to the Oracle8*i* database as Oracle database user SCOTT.	

Follow the steps below to grant external roles with Windows NT:

- Step 1: Perform Authentication Tasks on the Oracle8i Database Server
- Step 2: Perform Authentication Tasks on the Client Computer

Step 1: Perform Authentication Tasks on the Oracle8*i* Database Server
To perform authentication tasks on the Oracle8*i* database server:

1. Add the OS_ROLES initialization parameter to the INIT.ORA file.

2. Set OS_ROLES to TRUE.

    The default setting for this parameter is FALSE.

3. Ensure that you have the following line in your *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN\SQLNET.ORA file:

    ```
    SQLNET.AUTHENTICATION_SERVICES = (NTS)
    ```

4. Start SQL*Plus:

    ```
    C:\> SQLPLUS
    ```

5. Connect to your Windows NT server:

```
SQL> CONNECT INTERNAL
```

6. Create a new database role:

```
SQL> CREATE ROLE DBSALES3 IDENTIFIED EXTERNALLY;
```

where DBSALES3 is the name of the role for these steps. Substitute a role name appropriate to your database environment.

7. Grant Oracle roles to DBSALES3 that are appropriate to your database environment:

```
SQL> GRANT DBA TO DBSALES3 WITH ADMIN OPTION;

SQL> GRANT RESOURCE TO DBSALES3 WITH ADMIN OPTION;

SQL> GRANT CONNECT TO DBSALES3 WITH ADMIN OPTION;
```

8. Connect to the database with the INTERNAL DBA name:

```
SQL> CONNECT INTERNAL
```

9. Shut down the database:

```
SQL> SHUTDOWN
```

10. Restart the database:

```
SQL> STARTUP
```

11. Open the Windows NT User Manager.

12. Choose New Local Group from the User menu.

The *New Local Group* dialog box appears:
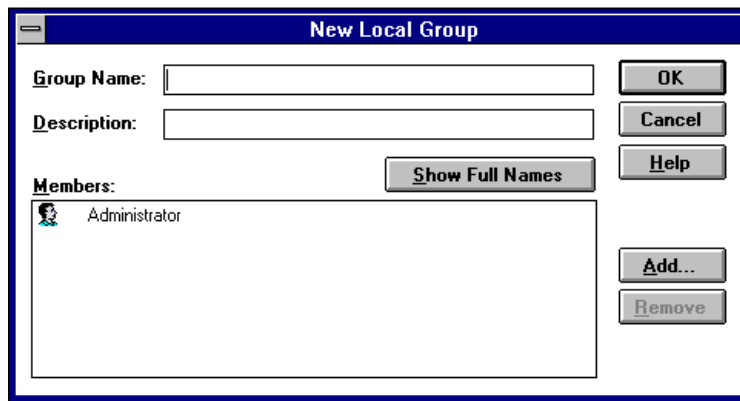
13. Enter the Windows NT local group name corresponding to the database role in the Group Name field with the following syntax:

ORA_*SID_ROLENAME* [_D] [_A]

where:

SID	Indicates the database instance.
ROLENAME	Identifies the database role granted to users of a database session.
D	Optional character indicating that this database role is to be the default role of the database user. If specified, this character must be preceded by an underscore.
A	Optional character indicating that this database role includes the ADMIN OPTION. This enables the user to grant the role to other roles only. If specified, this character must be preceded by an underscore.

For this example, ORA_ORCL_DBSALES3_D is entered.

14. Click Add.

The *Add Users and Groups* dialog box appears:
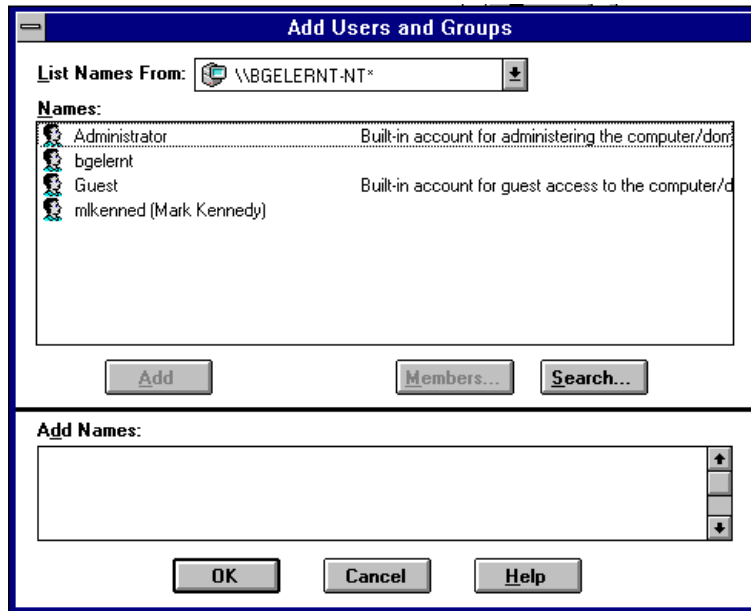
15. Select the appropriate Windows NT local or domain user name and click Add.

16. Click OK.

Your selection is added to the Members field of the *New Local Group* dialog box:
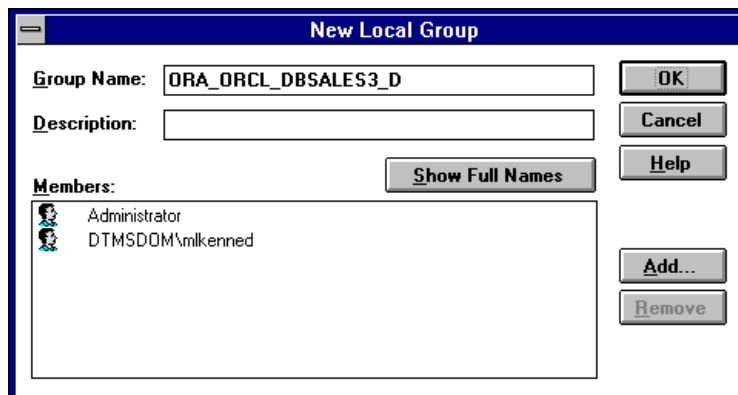
You can convert additional database roles to several possible Windows NT groups, as shown in the following table. Then, users connecting to the ORCL instance in this example and authenticated by Windows NT as members of these Windows NT local groups have the privileges associated with DBSALES3 and DBSALES4 by default (because of the _D option). DBSALES1 and DBSALES2 are available for use by the user if they first connect as members of DBSALES3 or DBSALES4 and use the SET ROLE command. If a user tries to connect with DBSALES1 or DBSALES2_A without first connecting with a default role, they are unable to connect. Additionally, users can grant DBSALES2 and DBSALES4 to other roles.

Database Roles	Windows NT Groups
DBSALES1	ORA_ORCL_DBSALES1
DBSALES2	ORA_ORCL_DBSALES2_A
DBSALES3	ORA_ORCL_DBSALES3_D
DBSALES4	ORA_ORCL_DBSALES4_DA

Note: When the Oracle8*i* database converts the group name to a role name, it changes the name to uppercase.

17. Click OK.

18. Exit User Manager.

Step 2: Perform Authentication Tasks on the Client Computer

To perform authentication tasks on the client computer:

1. Create a Windows NT local or domain user name with the same user name and password that exist on the Windows NT server (if the appropriate user name does not currently exist).

2. Ensure that you have the following line in your *ORACLE_BASE* \ *ORACLE_HOME*\NETWORK\ADMIN\SQLNET.ORA file:

```
SQLNET.AUTHENTICATION_SERVICES = (NTS)
```

3. Use Net8 Assistant to configure a network connection from your client computer to your Oracle8*i* database. See *Net8 Administrator's Guide* for instructions.

4. Start SQL*Plus:

```
C:\> SQLPLUS
```

5. Connect to the correct instance:

```
SQL> SET INSTANCE NET_SERVICE_NAME
```

where *NET_SERVICE_NAME* is the Net8 service name for the Oracle8*i* database that you created in Step 3.

6. Connect to the Oracle8*i* database:

```
SQL> CONNECT SCOTT/TIGER
```

You are connected to the Windows NT server over Net8 with the Oracle user name SCOTT/TIGER. The roles applied to the Oracle user name SCOTT consist of all roles defined for the Windows NT user name that were mapped to the database roles above (in this case, ORA_DBSALES3_D). All roles available under an authenticated connection are determined by the Windows NT user name and the Oracle-specific Windows NT local groups to which the user belongs (for example, ORA_*SID*_DBSALES1 or ORA_*SID*_DBSALES4_DA).

Administering Enterprise Users and Roles

Use Oracle Enterprise Security Manager to create and manage enterprise users, roles, and domains. Oracle Enterprise Security Manager is included as an integrated application of the Oracle Enterprise Manager Console. See the *Oracle Advanced Security Administrator's Guide* for more information on using Oracle Enterprise Security Manager.

> **Note:** You can administer external users and roles in Windows 2000 domains, but you cannot use Oracle Enterprise Security Manager to perform this administration. See "Administering External Users and Roles" on page 8-14 for more information on tools available for administering external users and roles.

9

Monitoring a Database

This chapter describes how to monitor Oracle8*i* for Windows NT.

Specific topics discussed are:

- Database Monitoring Overview
- Using Oracle Performance Monitor for Windows NT
- Using the Event Viewer
- Using Trace and Alert Files
- Viewing Threads Using the Oracle Administration Assistant for Windows NT

Database Monitoring Overview

The following tools enable you to monitor your Oracle8*i* database:

This Tool	Enables You To...
Oracle Performance Monitor for Windows NT	Monitor database objects, such as CPU usage, buffer cache, and background processes.
Event Viewer	Monitor database events.
Trace Files	Record occurrences and exceptions of database operations.
Alert Files	Record important information about error messages and exceptions during database operations.
Oracle Enterprise Manager Database Management Packs	Monitor and tune using tools with real-time graphical performance information. **See:** Your Oracle Enterprise Manager documentation set for more information.
Oracle Administration Assistant for Windows NT	View information on or kill Oracle threads.

Each tool is described in the following sections.

> **Additional Information:** See *Oracle8i Tuning and Performance* for general tuning information and your operating system documentation for additional information on Windows NT Performance Monitor results and optimizing database performance.

Using Oracle Performance Monitor for Windows NT

Oracle Performance Monitor for Windows NT is a graphical tool for measuring the performance of Oracle8*i* for Windows NT objects on a local server or other servers on a network. This tool is the same in appearance and operation as the Windows NT Performance Monitor, except it has been preloaded with Oracle8*i* database performance elements.

On each computer, you can view the behavior of objects, such as the buffer cache, data dictionary cache, data files, threads, and processes. An *object* is a graphical representation of an element in your system. Every element, resource, and device in your system can be represented as an object.

There is a set of counters associated with each object. A *counter* is a unit of measurement used by the Performance Monitor to display activity. The type of activity the counter measures is dependent upon the type of object.

Certain object types and their respective counters are present on all systems. Other counters, such as application-specific counters, appear only if the computer is running the associated software.

Each of these objects has an associated set of counters that provide information about device usage, queue lengths, delays, and information used to measure throughput and internal congestion.

Registry Information

When you install Oracle Performance Monitor for Windows NT, values are automatically set in the registry as described in Appendix C, "Oracle8i Configuration Parameters and the Registry".

The Oracle Performance Monitor for Windows NT allows you to monitor only one database instance at a time. For this reason, the registry contains the following values:

- Hostname

- Username

- Password

Use OPERFCFG to change these values. Oracle Corporation recommends setting the security level on each of these registry values.

> **See :**   Appendix C, "Oracle8i Configuration Parameters and the Registry", Modifying Oracle Performance Monitor for Windows NT Parameters, on how to use OPERFCFG.

To use Oracle Performance Monitor for Windows NT for another database instance on the same computer or a UNIX computer, change the values appropriately in the registry. You can also monitor non-NT Oracle databases by changing the Hostname registry value so it points to another computer specified in the TNSNAMES.ORA file.

Accessing Oracle Performance Monitor for Windows NT

To access Oracle Performance Monitor for Windows NT:

Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Performance Monitor for Windows NT.

The *Performance Monitor* window appears with the *Chart View*:

Counters   Objects
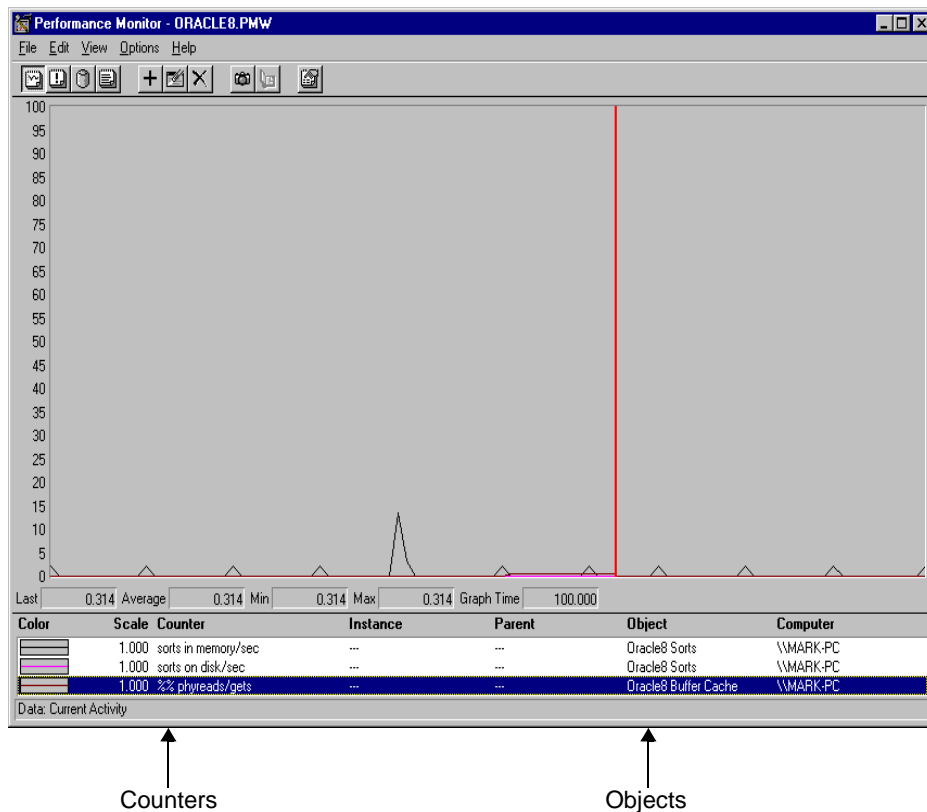
The Oracle Performance Monitor for Windows NT has four views you can choose from the View menu:

View	Description
Chart View	Displays database activity in real-time.
Alert View	Lets you know when certain minimum performance criteria are not being met, or maximum criteria are being exceeded.

View	Description
Log View	Maintains continuous records on performance.
Report View	Saves information about specific criteria.

> **Additional Information:** For complete information about the four views, see your Windows NT documentation.

Monitoring Oracle8*i* Objects

For each view (Chart, Alert, Log, and Report), you can decide on the objects you want to monitor and save those settings to a file. When an object is chosen, it is assigned a counter, a color, and added to the status bar at the bottom of Oracle Performance Monitor for Windows NT.

To add objects to a view:

1. Choose Add To (Chart, Alert, Log, Report) from the Edit menu.

   The Add to (Chart, Alert, Log, Report) dialog box appears.

   Below is the Add to Chart dialog box. Note the corresponding dialog boxes for the other views are different.
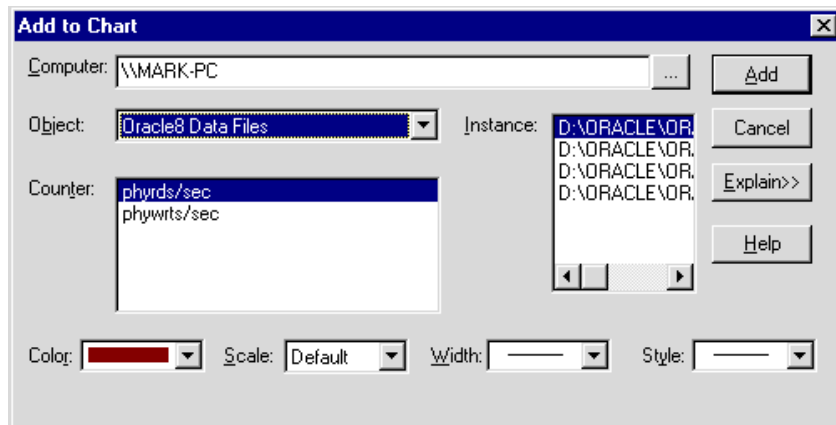
2. Select the objects you want to monitor, then click Add.

Below are the elements of the Add to Chart dialog box. The other views' dialog boxes have similar features.

Element	Description
Computer list box	Specify the computer you want to monitor.
Object drop-down list box	Select an object to monitor.
	Note: If no data or Oracle8i objects appear, either the database is not running, or an invalid host string or password has been entered. If the database is not started, exit Oracle Performance Monitor for Windows NT, start the database, and restart Oracle Performance Monitor for Windows NT.
Counter list box	Select a counter (or multiple counters) for the object you have selected. Note that the contents of the Counter box change depending upon your selection in the Object box.
	If you want details on how a counter works, highlight the counter and choose Explain.
Instance box	Select an instance for this counter.
Color box	Choose a color for the display of the selected counter.
Scale box	Choose the scale at which you want to display the counter.
Width box	Specify the width of the line on the graph.
Style box	Choose a different style for your graph line.

3. Click Done when you are finished.

The selections you have chosen to monitor are displayed.

Understanding Oracle Performance Objects

All Oracle8i system resources that can be monitored through Oracle Performance Monitor for Windows NT begin with *Oracle8i*. These measures are defined in *ORACLE_BASE\ORACLE_HOME*\DBS\PERF.ORA. The following table shows the Oracle8i objects and their associated counters. For additional information on these objects, see *Oracle8i Tuning and Performance*.

> **Note:** You can only monitor one instance at a time using Oracle Performance Monitor for Windows NT on a given computer.

Object	Counter	Description
Oracle8*i* Buffer Cache	phyrds/gets %	The percentage of phyrds/gets is calculated as a Miss ratio. The lower the Miss counter, the better. To improve performance, increase the number of buffers in the buffer cache, if memory is available on the machine. To make the buffer cache larger, increase the value of the DB_BLOCK_BUFFERS initialization parameter.
		This value is not time-derived.
Oracle8*i* Redo Log Buffer	redo log space requests	The value of this counter must be near zero. If this value increments consistently, processes have had to wait for space in the redo log buffer. In this case, it may be necessary to increase the size of the redo log buffer.
Oracle8*i* Data Dictionary Cache	getmisses/gets %	The value of this counter must be less than 10 or 15% for frequently accessed data dictionary caches. If the ratio continues to increase above this threshold while your application is running, increase the amount of memory available to the data dictionary cache.
		To increase the memory available to the cache, increase the value of the initialization parameter SHARED_POOL_SIZE. (See *Oracle8i Tuning and Performance* for more detailed information on tuning memory allocation in the Oracle8*i* database.)
		This value is not time-derived.
Oracle8*i* Library Cache	reloads/pins %	The percentage of SQL statements, PL/SQL blocks, and object definitions that required reparsing. Total Reloads must be near zero. If the ratio of Reloads to Pins is greater than 1%, then reduce the library cache misses.
		This value is not time-derived.
Oracle8*i* Data Files	■ phyrds/sec ■ phywrts/sec	Disk contention occurs when multiple processes try to access the same disk simultaneously. There are many ways of reducing disk contention, depending on the results from monitoring disk activity. Some corrective actions include: ■ distributing I/O ■ separating data files and redo log files ■ separating tables and indexes ■ striping table data These values are time-derived.
Oracle8*i* DBWR stats1		These counters are helpful in tuning the Buffer Cache.

Object	Counter	Description
	■ buffers scanned/sec	Buffers scanned/sec is the number of buffers the DBWR scanned per second. The buffers scanned are on the LRU (Least Recently Used) list.
	■ LRU scans/sec	LRU_scans/sec is the number of times the DBWR scanned the (Least Recently Used) buffer list per second.
Oracle8*i* DBWR stats2		These counters are helpful in determining how much work the DBWR has been requested to perform.
	■ timeouts/sec	Timeouts/sec is the number of times the DBWR timed-out per second. The DBWR is on a three second timeout interval. If the DBWR has not been posted within a three second interval, it times out.
	■ checkpoints/sec	Checkpoints/sec is the number of checkpoint messages processed by the database writer per second. Whenever a checkpoint occurs, the DBWR must be messaged (posted) to "write dirty buffers to disk".
Oracle8*i* Dynamic Space Management	recursive calls/sec	Dynamic extension causes Oracle8*i* to execute SQL statements in addition to those SQL statements issued by user processes. These SQL statements are called recursive calls. If Oracle8*i* makes excessive recursive calls while an application is running, it may be necessary to determine the cause. Examine the recursive calls statistic through the dynamic performance table V$SYSSTAT.
Oracle8*i* Free List	free list waits/ requests %	Contention for free lists is reflected by contention for free data blocks in the buffer cache. You can determine if contention for free lists is reducing performance by querying V$WAITSTAT. If the number of free list waits for free blocks is greater than 1% of the total number of requests, consider adding more free lists to reduce contention.
Oracle8*i* Sorts	■ sorts in memory/sec ■ sorts on disk/sec	The default sort area size is adequate to hold all the data for most sorts. However, if your application often performs large sorts on data that does not fit into the sort area, then you may want to increase the sort area size.

Oracle Performance Monitor for Windows NT Troubleshooting Information

If no data or Oracle8*i* objects appear in the Objects list of the Add to Chart dialog box, either:

- the database is not running

- an invalid host string or password has been entered in the registry

The OPERF.LOG file located in *ORACLE_BASE\ORACLE_HOME*\DBS contains error messages about Oracle Performance Monitor for Windows NT.

To resolve this problem:

1. Check the *ORACLE_BASE\ORACLE_HOME*\DBS\OPERF.LOG file for error messages.

2. Resolve the problem as follows:

   - If the log file indicates an invalid host string or password, check the registry for correct values for Hostname, Password, and Username. See section "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet" on page C-9 for further information about these values.

   - If the database is not started, exit Oracle Performance Monitor for Windows NT and restart the database.

3. Restart Oracle Performance Monitor for Windows NT.

Using the Event Viewer

Oracle8*i* for Windows NT problems and other significant occurrences are recorded as events. These events are recorded in an application event log. View and manage these recorded events in the Event Viewer.

Accessing the Event Viewer

To access the Event Viewer:

1. Choose Start > Programs > Administrative Tools > Event Viewer.

   The Event Viewer window appears.

2. Choose Application from the Log menu.

The Application view window displays the following information:

The date event took place.

The time event took place.

Application that recorded event.

Classification of event.

Unique number assigned to event.

Computer name on which event occurred.
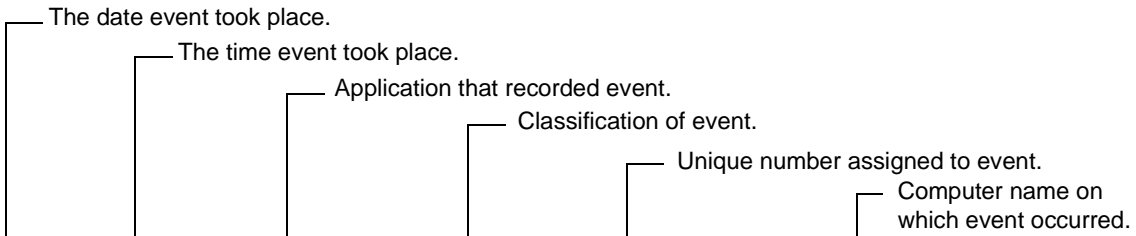
	Date	Time	Source	Category	Event	User	Computer
⊗	7/26/98	9:34:27 AM	Perflib	None	1011	N/A	BGELERNT-NT
⊗	7/26/98	9:34:27 AM	Perflib	None	1011	N/A	BGELERNT-NT
⚠	7/24/98	4:48:22 PM	Oracle.orcl	None	16	N/A	BGELERNT-NT
⚠	7/24/98	4:48:21 PM	Oracle.orcl	None	12	N/A	BGELERNT-NT
ⓘ	7/24/98	4:48:18 PM	Oracle.orcl	None	8	N/A	BGELERNT-NT
ⓘ	7/24/98	4:48:18 PM	Oracle.orcl	None	34	N/A	BGELERNT-NT
ⓘ	7/24/98	3:06:59 PM	Oracle.orcl	None	34	N/A	BGELERNT-NT
ⓘ	7/23/98	12:23:09 PM	Oracle.orcl	None	34	N/A	BGELERNT-NT
ⓘ	7/23/98	12:08:14 PM	Oracle.orcl	None	5	N/A	BGELERNT-NT
ⓘ	7/23/98	12:08:14 PM	Oracle.orcl	None	5	N/A	BGELERNT-NT
ⓘ	7/23/98	12:08:14 PM	Oracle.orcl	None	5	N/A	BGELERNT-NT
ⓘ	7/23/98	12:08:13 PM	Oracle.orcl	None	5	N/A	BGELERNT-NT
ⓘ	7/23/98	12:08:13 PM	Oracle.orcl	None	5	N/A	BGELERNT-NT
ⓘ	7/23/98	12:08:13 PM	Oracle.orcl	None	5	N/A	BGELERNT-NT
ⓘ	7/23/98	12:08:12 PM	Oracle.orcl	None	5	N/A	BGELERNT-NT

Event Viewer - Application Log on \\BGELERNT-NT

Log View Options Help

Reading the Event Viewer

The icons beside each event determine the type of event.

Icon	Event Type	Description
red (stop sign)	Error	Indicates an error. Always check these icons.
blue (informational)	Information	Indicates a non-critical system event. You can ignore these icons unless you want to track a specific event.
yellow (exclamation point)	Warning	Indicates a special event, such as the termination of an instance or the shutdown of services. Investigate these icons, but they are usually non-critical.

Oracle8*i* for Windows NT events display with a source of Oracle.orcl. Oracle.orcl consists of the following event IDs:

Event ID	Description
IDs other than 34	Specifies general database activities, such as an instance being started or stopped.
34	Specifies an audit trail event. These events are recorded if the AUDIT_TRAIL parameter is set to DB (TRUE) or OS in the INIT.ORA file.
	The OS option enables system-wide auditing and causes audited records to be written to the Event Viewer.
	The DB option enables system-wide auditing and causes audited records to be written to the database audit trail (the SYS.AUD$ table). Some records, however, are written to the Event Viewer.

Using the Event Viewer

To use the Event Viewer:

1. Look at the icons.

2. Double-click an icon to analyze (especially red icons).

   The Event Detail dialog box appears with more information about the selected event:

A text description of the event.

Click Bytes to see the information in hexadecimal format. Or, click Words to see DWORDS for the same data.
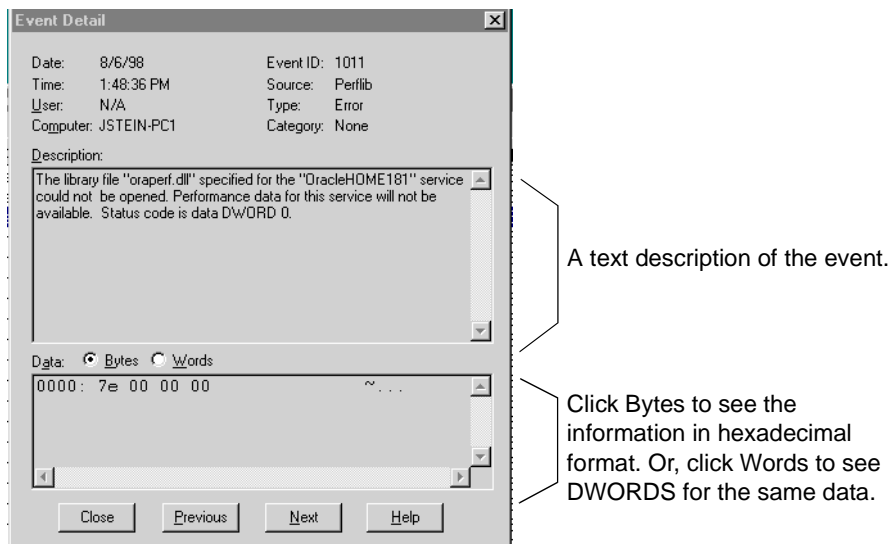
See Also: Microsoft Windows NT documentation for more information on using the Windows NT Event Viewer.

Managing the Event Viewer

Setting AUDIT_TRAIL to DB or OS causes more records to be written to the Event Viewer. This can fill up the Event Viewer log file. Follow these procedures to increase the log file size.

To increase log file size:

1. Choose Log Settings from the Log menu.

    The Event Log Settings dialog box appears.

2. Adjust the setting in the Maximum Log Size field to an appropriate level.

3. Click OK.

    You are returned to the Event Viewer.

> **Warning:** Audit information cannot be spooled to a file. The **AUDIT_FILE_DEST parameter is not supported in Windows NT and should not be added to the INIT.ORA file.**

Using Trace and Alert Files

Oracle8*i* for Windows NT background threads use trace files to record occurrences and exceptions of database operations, as well as errors. Background thread trace files are created regardless of whether the BACKGROUND_DUMP_DEST parameter is set in the INIT.ORA initialization parameter file. If BACKGROUND_DUMP_DEST is set, the trace files are stored in the directory specified. If the parameter is not set, the trace files are stored in the *ORACLE_BASE*\ADMIN*DB_NAME*\BDUMP directory.

Oracle8*i* database creates a different trace file for each background thread. The name of the trace file contains the name of the background thread, followed by the extension.TRC. Sample trace file syntax includes:

- *SID*DBWR.TRC
- *SID*SMON.TRC

where *SID* represents the name of the instance.

Trace files are also created for user threads if the USER_DUMP_DEST parameter is set in the initialization parameter file. The trace files for the user threads have the form ORA*xxxxx*.TRC, where *xxxxx* is a 5-digit number indicating the Windows NT thread ID.

The alert file contains important information about error messages and exceptions that occur during database operations. Each Oracle8*i* for Windows NT instance has one alert file; information is appended to the file each time you start the instance. All threads can write to the alert file.

For example, when automatic archiving of redo logs is halted because no disk space is available, a message is placed in the alert file. The alert file is the first place to check if something goes wrong with the database and the cause is not immediately obvious.

The alert file is named *SID*ALRT.LOG and is found in the directory specified by the BACKGROUND_DUMP_DEST parameter in the INIT.ORA initialization parameter file. If the BACKGROUND_DUMP_DEST parameter is not set, the *SID*ALRT.LOG file is generated in *ORACLE_BASE*\ADMIN*DB_NAME*\BDUMP. Alert files should be deleted or archived periodically.

Viewing Threads Using the Oracle Administration Assistant for Windows NT

To view information on Oracle threads using the Oracle Administration Assistant for Windows NT:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Administration Assistant for Windows NT.

2. Right-click the *SID*, where *SID* is a specific instance name, such as ORCL.

3. Choose Process Information.

4. View information on appropriate threads. If you want to kill a thread select the thread you want to kill. Click Kill Thread.

10
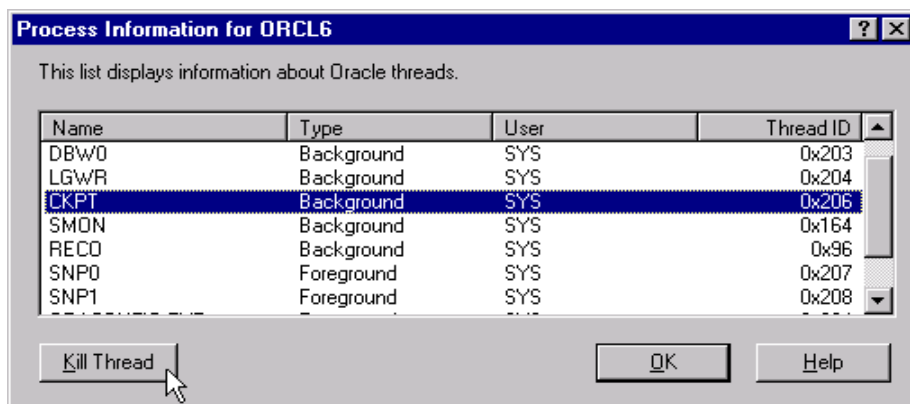
Tuning Windows NT to Optimize Oracle8*i*

This chapter describes how to tune the Windows NT Server operating system to ensure that your Oracle8*i* database is running in the best possible environment.

Specific topics discussed are:

- Overview
- 4 GB RAM Tuning (4 GT) for Windows NT Server, Enterprise Edition
- Oracle8i Support for the Intel Extended Server Memory Architecture (ESMA)
- Reduce Priority of Foreground Applications on the Server Console
- Configure Windows NT Server to Be an Application Server
- Disable Unnecessary Services
- Remove Unused Network Protocols
- Reset the Network Protocol Bind Order
- Apply Latest Reliable Windows NT Server Service Pack
- Use Hardware or Operating System Striping
- Span Windows NT Server Virtual Memory Paging File Across Physical Volumes
- Close All Unnecessary Foreground Applications

Overview

Compared to UNIX, Windows NT Server offers considerably fewer *knobs* that can be adjusted to tune the operating system (OS). This reduces the ability of systems administrators to optimize Windows NT Server performance, but helps to make Windows NT Server easier to use than some operating systems.

There are still ways, however, to make Windows NT Server a better application server environment for the Oracle8*i* database. Most of these OS-specific procedures have the effect of reserving more system resources for the Oracle8*i* database, such as CPU, memory, and disk I/O. These procedures are described in this chapter. In addition, the Oracle8*i* database is a high-performance database management system that effectively uses the resources within your Windows NT computer. In general, the Windows NT computer that is running your Oracle8*i* database should *not* also serve as any of the following:

- primary or backup domain controller
- file or print server
- remote access server
- router

These configurations consume both network, memory, and CPU resources. In addition, the Windows NT computer that is running your Oracle8*i* database should not be locally accessed with a high frequency or intensively used for local user processing, unless there exist significant resources to accommodate all this activity.

> **Note:** The information described in this chapter is specific to Oracle8*i* database installations on Windows NT Server, and not on Windows NT Workstation. This information is also applicable to Oracle8*i* database installations on Windows NT Server 4.0, Enterprise Edition. Windows NT Server 4.0, Enterprise Edition includes the capability for using additional products, such as the Microsoft Cluster Server, which is required if you want to implement Oracle Fail Safe (OFS).

4 GB RAM Tuning (4 GT) for Windows NT Server, Enterprise Edition

Windows NT Server, Enterprise Edition includes a new capability called 4 GB RAM Tuning (4GT). This capability allows memory-intensive applications running on Windows NT Server, Enterprise Edition to utilize up to 50% more RAM on Intel Architecture servers. It does this by reducing the potential RAM allocated to the NT kernel from 2 Gigabytes (GB) to 1 GB, and increasing the potential RAM allocated to processes from 2 GB to 3 GB. Note that this capability is not currently available on any other Microsoft operating system.

For Oracle Server releases 7.3.4 or later, there is no additional Oracle configuration required to take advantage of 4 GB RAM Tuning, nor is there a requirement that this feature be enabled in order to use Oracle Server.

> **Additional Information:** See your Microsoft operating system documentation for more information on using 4 GB RAM Tuning, or visit:
>
> `http://www.microsoft.com/ntserver/ntserverenterprise/ex`
> `ec/feature/4gbt.asp`

Oracle8*i* Support for the Intel Extended Server Memory Architecture (ESMA)

A new feature in Oracle8*i* for Windows NT is support for Intel ESMA, which allows Oracle8*i* to access more than the 4 GB of RAM traditionally available to Windows NT applications.

> **Note:** This feature is only available on Intel Pentium II and Pentium III Xeon 32-bit processors.

Specifically, Oracle8*i* (in conjunction with Intel's PSE36 driver) can now allocate substantially more database buffers than previous releases. Further details are posted at:

`http://www.intel.com/ebusiness/server/resources/pentiumii/xeon/esma.pdf`

To take advantage of this support, you must do the following:

1. More than 4 GB of RAM must be present in the server on which Oracle8*i* will run.

2. Windows NT v4.0 Enterprise Edition, Service Pack 3 or later must be installed.

3. The Intel PSE36 driver must be installed and operational. See

   http://developer.intel.com/vtune/pse36/index.htm

   for further PSE36 system requirements and for download instructions.

4. USE_INDIRECT_DATA_BUFFERS=TRUE must be present in the INIT.ORA for the database instance that will use the PSE36 driver. If this parameter is not set, then Oracle8*i* behaves in exactly the same way as previous releases.

5. Set DB_BLOCK_BUFFERS and DB_BLOCK_SIZE as desired for the database. Note that the total number of bytes of database buffers (that is, DB_BLOCK_ BUFFERS multiplied by DB_BLOCK_SIZE) is no longer limited to 3 GB as was the case in previous releases.

6. The VLM_BUFFER_MEMORY registry parameter must be created and set in the appropriate key for your Oracle home in the Windows NT Registry. This parameter is specified in bytes and has a default of 1 GB. When set, this parameter tells Oracle8*i* how much non-PSE36 memory to use for database buffers. This memory comes from Oracle8*i*'s virtual address space, as was the case in previous releases. Setting this parameter to a large value has the effect of using more of Oracle8*i*'s address space for buffers and using less PSE36 memory for buffers. However, since accessing PSE36 buffers is somewhat slower than accessing virtual address space buffers, tune this parameter to be as large as possible without adversely limiting database operations.

   For instance, assume that the Oracle8*i* database is running on a machine with 8 GB of RAM, which means that the PSE36 driver has control of 4 GB of RAM. If DB_BLOCK_BUFFERS=2500000 and DB_BLOCK_SIZE=2048, then a total of 5 GB of database buffers needs to be allocated. If VLM_BUFFER_MEMORY is set to 1 GB, then 1 GB of buffers come from the Oracle8*i* virtual address space and 4 GB come from the PSE36 driver. If you set VLM_BUFFER_MEMORY to 500 MB, an error occurs at startup because there is not 4.5 GB of memory available to the PSE36 driver for database buffers. Likewise, if you set VLM_BUFFER_ MEMORY to 3 GB, an error occurs because the Oracle8*i* address space is limited to 3 GB on Windows NT, and this address space must also hold Oracle8*i* code, shared pool, PGA memory, and other structures. In general, the higher the VLM_BUFFER_MEMORY is set, the fewer connections and memory allocations will be possible for Oracle8*i*. The lower VLM_BUFFER_MEMORY is set, the slower the performance will be.

7. Once these parameters are set, the Oracle8*i* database can be started up and will function exactly the same as before except that more database buffers are available to the instance. In addition, disk I/O may be reduced since more

Oracle data blocks can be cached in the SGA. If out of memory errors occur during the startup sequence, verify the following:

a. PSE36 driver is installed and functional

b. DB_BLOCK_BUFFERS is not set too high for the amount of memory in the machine. Note that more memory than just the database buffers themselves is required when starting up the database. For each database buffer, a database buffer header is also allocated in Oracle8i's virtual address space. When allocating 2,000,000 database buffers, the memory for these buffer headers amounts to several hundred megabytes. This must be considered when setting DB_BLOCK_BUFFERS and VLM_BUFFER_MEMORY.

c. VLM_BUFFER_MEMORY is not set too high for the amount of address space available to Oracle8*i*. In Windows NT's Performance Monitor, under the Process object, monitor the Virtual Bytes counter for the "ORACLE" process. If this counter approaches 3 GB, then out of memory errors can occur. If this happens, reduce DB_BLOCK_BUFFERS and/or VLM_BUFFER_MEMORY until the database is able to start up.

8. Currently, there is a limitation in Server Manager for NT whereby the amount of database buffers displayed during database startup is incorrect if more than 4 GB of buffers are in use. For instance, if 5 GB of buffers are used, Server Manager will incorrectly report that 1 GB are being used. This limitation will be fixed in the next release of Oracle8*i*.

Reduce Priority of Foreground Applications on the Server Console

Interactive foreground applications running on Windows NT Server are given priority over background processes. This is the setting provided by default during the installation of Windows NT Server. In order to prevent foreground applications on the server console from taking excessive processor time away from the Oracle8*i* database, remove the priority for foreground applications.

To reduce the priority of foreground applications:

1. Choose Start > Settings > Control Panel > System.

   The *System Properties* dialog box appears.

2. Click the Performance tab.

   The *Performance* tab displays:
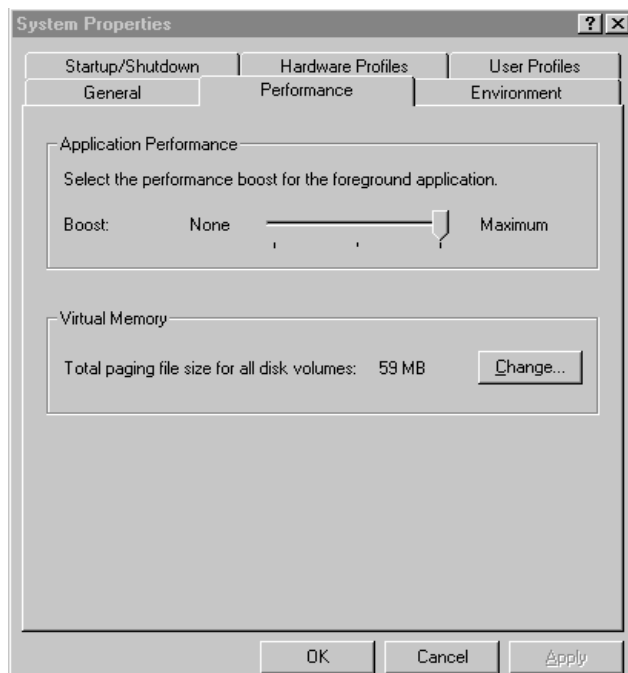
3. Move the Application Performance Boost slider to None.

4. Click OK.

5. Exit the Control Panel.

Configure Windows NT Server to Be an Application Server

Windows NT Server is a 32-bit OS capable of addressing a 4 GB memory space (that is, 2^{32} = 4,294,967,296 bytes = 4 GB). Half of this addressable memory space is reserved for system services and the file cache. The other half is addressable by user programs, such as the Oracle8*i* database.

Very few Windows NT Servers are equipped with 4 GB of RAM, so virtual memory plays an important role in Windows NT Server. The Windows NT Server memory manager tries to balance each application's usage of memory by dynamically paging memory between physical RAM and a virtual memory paging file. If an application is particularly memory-intensive (like the Oracle8*i* database) or if a large number of applications are running concurrently, the combined memory requirements of the applications may exceed the capacity of physical memory.

The Windows NT memory manager divides up system memory into three different pools.

Windows NT Server Memory Pools		
Category	**Pools**	**Approximate % of Total Memory**
System Area	Kernel and other system services	9%
	File Cache	41%
User Area	Paged Memory	50%

The large proportion of memory (41%) reserved for file caching can be quite beneficial to file and print servers but may not be advantageous to application servers that often run memory-intensive network applications. A Windows NT Server file cache is particularly unnecessary for the Oracle8*i* database, which performs its own caching (through the System Global Area Memory).

Windows NT Server is set by default to perform as a file and print server with a large file cache. Reset the server memory model for network applications so that the file cache is reduced and more physical memory is available for the Oracle8*i* database.

To configure Windows NT Server to be an applications server:

1. Choose Start > Settings > Control Panel > Network.

   The *Network* dialog box appears.

2. Click the Services tab.

   The following information appears:
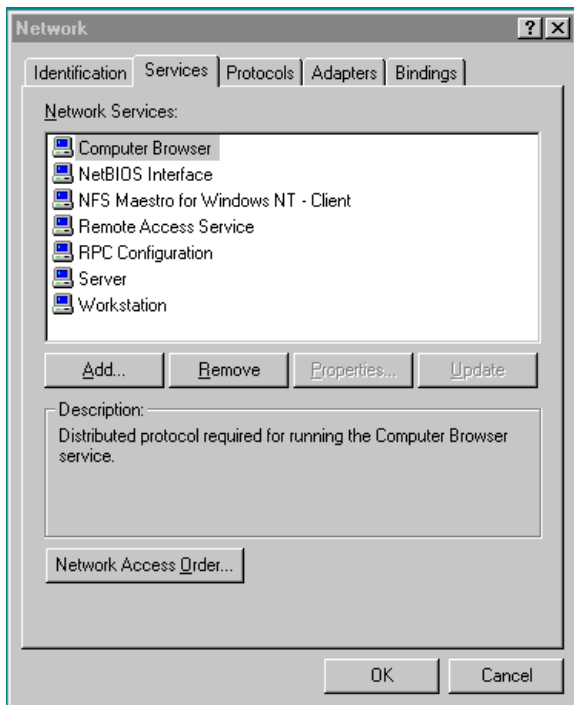
3. Select the Server service and click Properties.

4. Select the radio button for a network applications configuration and click OK.

5. Click OK on the *Network* dialog box.

6. Exit the Control Panel.

7. Reboot the system for the changes to take effect.

Disable Unnecessary Services

Once the file cache has been significantly reduced in size (as described in section "Configure Windows NT Server to Be an Application Server" on page 10-7), retrieve additional physical memory for the Oracle8*i* database by disabling services not needed for core OS functionality.

To disable unnecessary services:

1. Choose Start > Settings > Control Panel > Services.

   The *Services* dialog box appears.

2. Scroll through the list of services and identify any unnecessary services. This is a partial list of services that can or cannot be disabled:

You Can Disable These Services	Do Not Disable These Services
License Logging Service	Alerter
Plug and Play	Computer Browser
Remote Access Autodial Manager	EventLog
Remote Access Connection Manager	Messenger
Remote Access Server	OracleService*SID*
Telephony Service	Oracle*HOME_NAME*TNSListener
	OracleStart*SID* (for releases 8.0.5 and earlier)
	Remote Procedure Call (RPC) Service
	Server
	Spooler
	TCP/IP NetBIOS Helper
	Workstation

> **Note:** Consult with your systems administrator to find out if there are additional services that can be disabled.

3. Select the service.

4. Click Startup.

   The *Service* dialog box appears.

5. Select Disabled in the Startup Type field.

6. Click OK.

7. Exit the Control Panel.

Remove Unused Network Protocols

Remove all unnecessary network protocols on Windows NT Server so that processing time may be concentrated on servicing only critical protocols.

To remove unnecessary network protocols:

1. Choose Start > Settings > Control Panel > Network.

   The *Network* dialog box appears.

2. Click the Protocols tab.
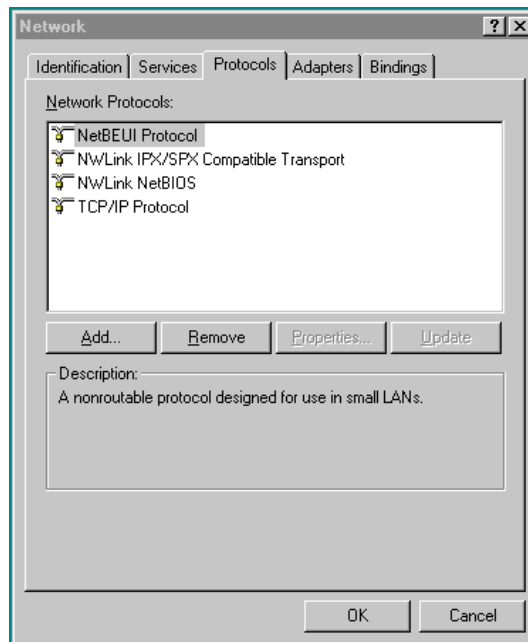
3. If multiple protocols are installed, choose one, preferably TCP/IP, for use as the sole network protocol, unless others are necessary for server functionality.

4. Select unneeded network protocols and click Remove.

5. Click Yes in the confirmation dialog box until only the necessary protocols remain.

6. Exit the Control Panel.

Reset the Network Protocol Bind Order

If multiple protocols must be installed on the server, then prioritize the bindings so that the protocol most frequently used by the Oracle8*i* database is given the highest priority.

To reset the network protocol bind order:

1. Choose Start > Settings > Control Panel > Network.

   The *Network* dialog box appears.

2. Click the Bindings tab.

   The *Bindings* tab displays:
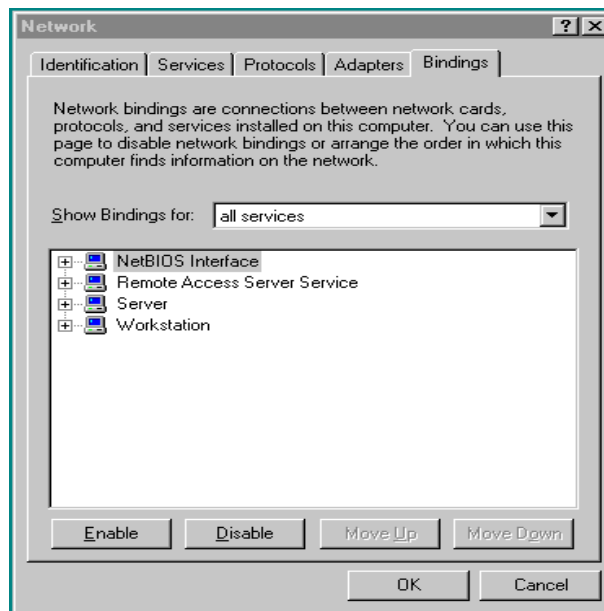
3. Select *all services* from the *Show Bindings for* drop-down list box.

4. Double-click Server to expand the list of currently installed services.

   All available protocol adapters display:
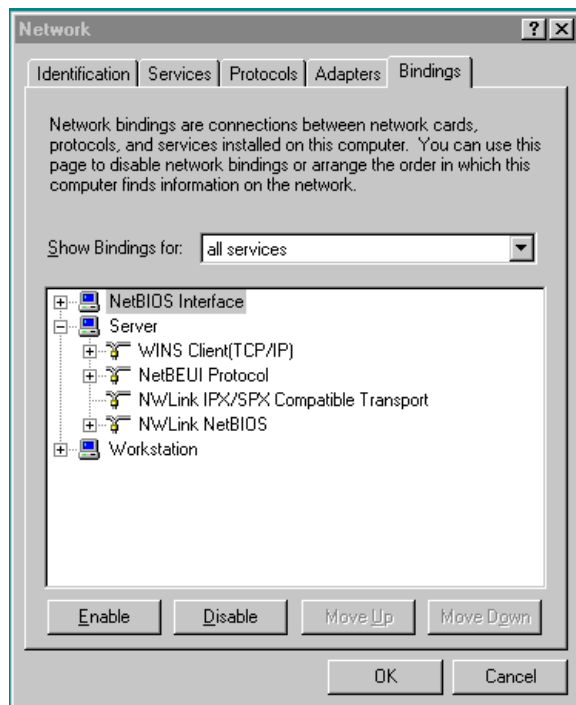
5. If the primary Oracle8*i* database protocol is not at the top of the list, select the protocol.

6. Click Move Up until the primary protocol is at the top of the list.

7. If multiple network interface cards (NICs) are installed, expand each protocol and move the NIC used most often by the Oracle8*i* database to the top of the list.

8. Click the OK button to save the changes.

9. Exit the Control Panel.

10. Reboot Windows NT Server to enable the new changes.

Apply Latest Reliable Windows NT Server Service Pack

Microsoft releases operating system patches, called Service Packs, on a quarterly basis. Service Packs are sometimes abbreviated as SP*x* where *x* is the release number of the Service Pack (for example, Windows NT Server 4.0 SP4 denotes a Windows NT Server 4.0 installation with Service Pack 4 applied). Service Packs are a collection of bug fixes and product enhancements to the basic Windows NT Server release. In general, apply Service Packs as soon as it is safe to do so, since they fix bugs and can improve Windows NT Server performance or functionality.

While the Service Packs are supposed to fix bugs, there have been reports (for example, the initial release of Windows NT Server 4.0 SP2) of bugs within the patch updates themselves. In general, it is safest to wait a few weeks after a Service Pack is released before implementing it. This allows time for other field sites to report any problems with the SP release.

The latest Windows NT Server Service Packs (as a self-extracting archive) may be downloaded from:

`http://support.microsoft.com/support/ntserver/content/servicepacks/default.asp`

To install a Service Pack:

1. Download the Service Pack version you want.

2. Read the README file, which contains important installation instructions.

   Unless there are assurances that the Service Pack works without flaws on Windows NT Server, choose to create an Uninstall directory. This enables the Service Pack to be removed and the original configuration to be restored.

   Service Pack files overwrite similarly-named files from those in the previous Windows NT Server configuration. However, Service Pack files can be overwritten in turn by setup programs that copy files from the original installation media.

   For example, installing a new network protocol or printer driver usually requires the copying of files from the original Windows NT Server installation media. When Service Pack files are comprehensively or selectively overwritten, the Service Pack must be re-applied.

To uninstall a Service Pack (if an Uninstall directory was created):

1. Launch the UPDATE.EXE (or the self-extracting archive) program.

2. Follow the appropriate prompts.

Use Hardware or Operating System Striping

Now that hard disk drives are relatively inexpensive, Windows NT Server should use logical volumes comprised of striped physical disks.

Compared to solid state CPU and memory speeds, mechanical hard disk drives are extremely slow. Data striping is an effective means of reducing the impact of relatively slow hard drives by distributing file I/O across a number of hard drives simultaneously.

Striping data across a number of disks is one example of a redundant array of inexpensive disks (RAID). There are several different types of RAID, ranging from high performance to high reliability. The three most common RAID levels in Oracle8*i* database installations are RAID-0, RAID-1, and RAID-5. Each are described below, along with a fourth RAID level, RAID 0+1:

RAID Level	Description	Read Penalty	Write Penalty
0 (Disk striping)	Enables high-performance, non-fault tolerant disk striping. Multiple physical hard disks are aggregated into a logical whole, either by a disk controller or through the operating system (for example, Windows NT Server stripe sets). Data operations against the logical volume are broken down into the "number of physical drives in the array" chunks, making simultaneous use of all disks. Given identical hard disks, if one hard disk has a throughput rate of DISKRATE operations/second, then a RAID 0 logical volume has a rate of: (DISKRATE * [number of physical drives in array]) operations/second The downside is that there is no fault tolerance and if one disk in the logical volume fails, the whole volume fails and must be restored from a backup.	1:1 (1 I/O per read request)	1:1 (1 I/O per write request)
1 (Disk mirroring)	Enables fault tolerant disk mirroring (some chance of a performance penalty). Essentially, every write to a mirrored disk is duplicated on another drive dedicated to this purpose (the mirror drive). If the mirrored disk fails, the mirror drive is brought online in real-time. After the faulted drive is replaced, the mirror configuration can be re-established.	1:1 (may benefit from split reads on some controllers) (For example, lower seek times when the controller knows which mirror contains the fastest accessed data.)	2:1 (writing to two sources)

RAID Level	Description	Read Penalty	Write Penalty
0+1	Enables mirroring of an array of striped hard disks. This is a blend of RAID 0 and RAID 1, offering high-performing fault tolerance.	1:1 (may benefit from split reads on some controllers) (For example, lower seek times when the controller knows which mirror contains the fastest accessed data.)	2:1
5 (Distributed Data Guarding (Disk striping with parity)	Enables distributed data guarding, eliminating the costly need to mirror. In RAID 5, multiple hard disks are aggregated into a striped logical volume, similar to RAID 0, but each drive contains parity information such that any single drive failure is tolerated. With one failed drive, a RAID-5 system may allow continued access to the data, although access times are greatly reduced due to the on-the-fly rebuilding of bytes from the parity information. RAID-5 solutions usually allow hot-swapping of faulty drives with replacements, triggering a rebuild of the failed drive's data onto the replacement from the parity information.	1:1	4:1 (2 reads/2 writes during parity calculation)

Data striping is achieved at the hardware level through the use of a RAID controller or, less efficiently, at the operating system level through Windows NT Server stripe sets. For a performance Windows NT Server configuration, data striping without parity (RAID-0 or equivalent) may be the best choice.

This example demonstrates the creation of a Windows NT Server stripe set. Suppose that there are six SCSI-2 hard disks attached to a common non-RAID disk controller. Each hard drive is not yet partitioned.

To create a Windows NT Server stripe set:

1. Choose Start > Programs > Administrative Tools (Common) > (Disk Administrator.

    The *Disk Administrator* dialog box appears.
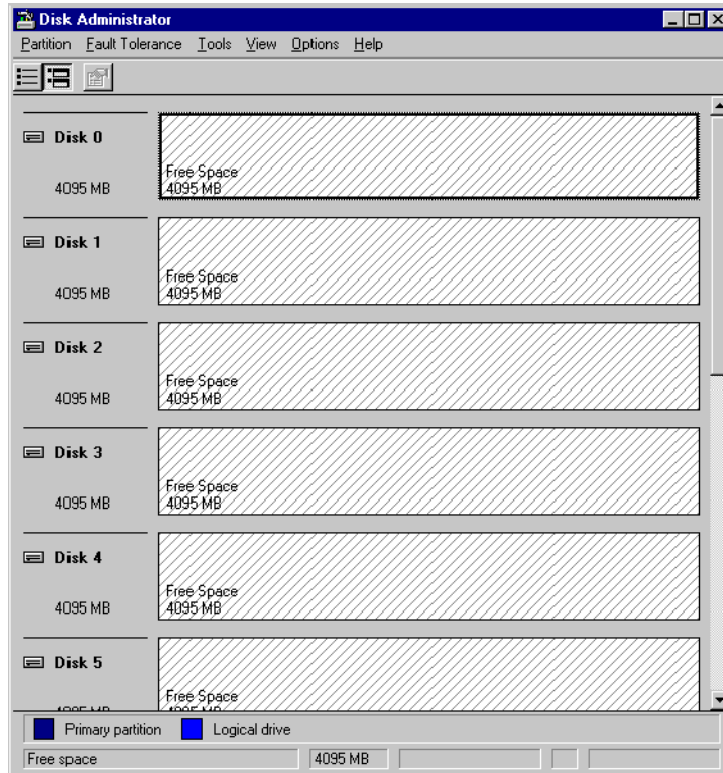
2. Click drive 0's free space.

3. Choose Create Extended from the Partition menu to create an extended partition sized to the hard disk's total capacity.

Note that the hash marks filling the partitioned drive 0's free space are now in the opposite direction to the marks in the free space of the unpartitioned hard disks:
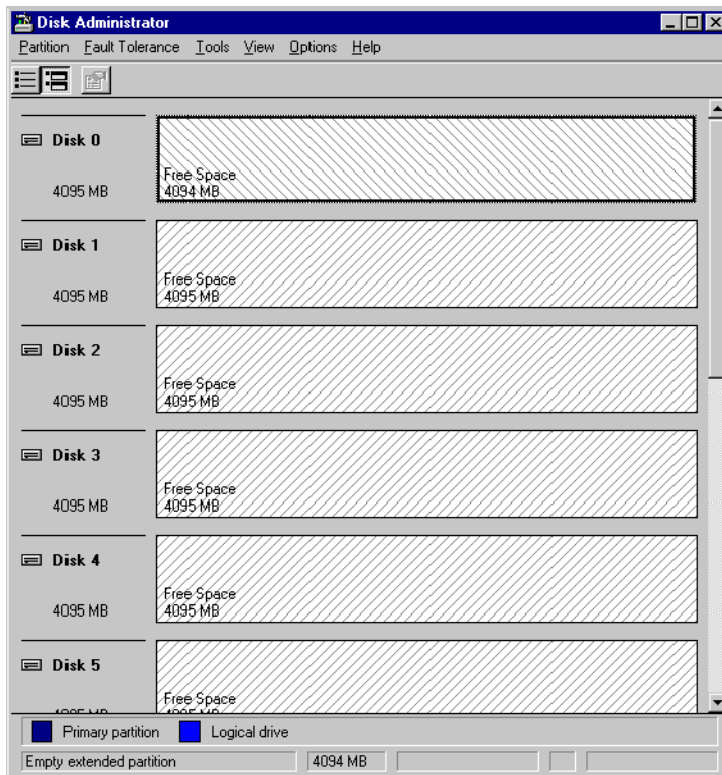
4. Repeat steps 2 and 3 for each of the five remaining unpartitioned hard disks.

5. Click the first (topmost) drive.

6. Press Ctrl while clicking the remaining five drives so that all six hard disks are selected.

7. Choose Create Stripe Set from the Partition menu.

8. Click OK on the *Logical Volume Size* dialog box. The maximum total size of the stripe set volume will be:

   [number of hard disks] * [capacity of the smallest selected partition]

   The Disk Administrator automatically adjusts the sizes of each drive's stripe if a stripe set size of less than the maximum allowable is chosen.

9. Choose Commit Changes Now from the Partition menu to save the volume changes.

   The logical stripe set volume across all physical hard disks appears.
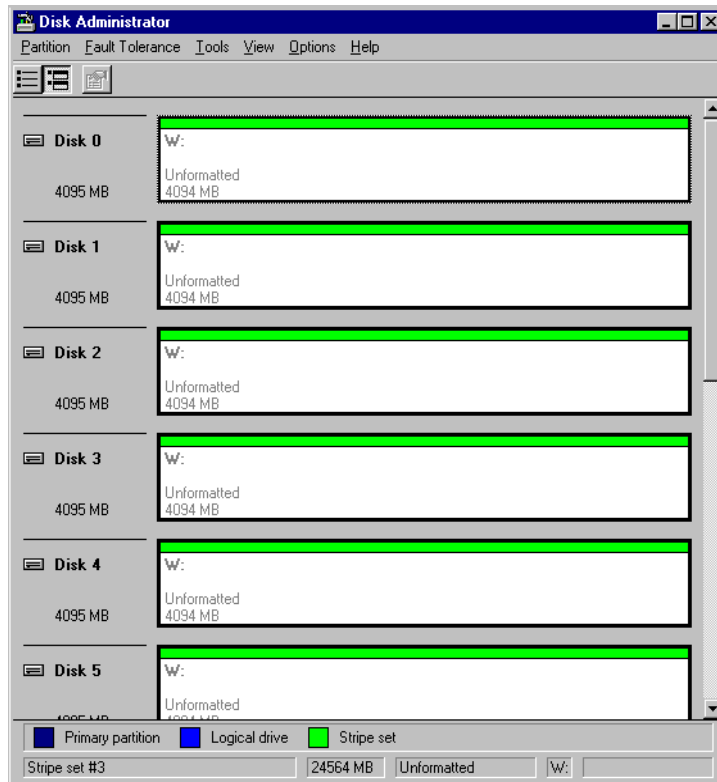
10. Reboot the system to enact the change. After restarting, the new logical stripe set volume is ready to be formatted.

.

> **Note:**   It is not strictly necessary to create an extended partition on each hard disk before creating a stripe set, but it is useful to do so for cases when a drive's stripe does not occupy its entire capacity and additional non-stripe set logical volumes are created on the drive. Extended partitions are necessary, however, when creating more than four logical partitions due to a Windows NT limit of only four primary (non-extended) partitions.

Multiple Striped Volumes for Sequential and Random Access

If there are enough physical disks in Windows NT Server, at least two striped volumes should be created (in addition to a stand-alone hard disk or striped volume for the OS). One striped volume can be used for sequential data access and the other can be used for random data access.

Oracle8*i* database redo logs and archived redo logs, for example, are written in sequential order. Because of the reduced head movement, hard disks perform best when reading or writing sequential data.

Oracle8*i* database data files, however, are usually accessed in random order. Random access in a hard disk results in significant head movement, translating to slower data access.

Unless redo logs are separated from data files (at the physical device level), undo file I/O contention may result, lowering the access times for both types of files.

Span Windows NT Server Virtual Memory Paging File Across Physical Volumes

Even if the Oracle8*i* database is the only network application running on Windows NT Server, some virtual memory paging is likely to happen as the Windows NT Server memory manager attempts to move an application's seldom-used pages to disk in order to free up more physical memory for *hot* pages.

Multiplexing the Windows NT Server virtual memory paging file is a good strategy to boost overall system performance. Splitting the paging file onto at least two different physical volumes (or logical volumes as long as the underlying physical volumes do not overlap) provides a significant performance boost to virtual memory swapping operations.

Even though this is a good technique to increase the speed of virtual memory paging, too much paging activity is still a performance hit and should be corrected by adding more RAM to the server.

General Page File Sizing Tip

It is recommended that the total combined size of the page files be at least the same size as the physical amount of RAM on the computer, and configurations where the combined size is two to four times the size of the physical RAM are not uncommon. Although the goal is to minimize paging as much as possible, a situation in which the operating system runs out of or low on paging space is to be avoided at all costs. Adequate paging files spaced across physical disks spread out the I/O most efficiently, since the operating system spreads paging evenly across page files.

Close All Unnecessary Foreground Applications

Once the procedures in the previous sections have been applied, remember to close any unnecessary foreground applications. Three are described below:

- Startup Folder
- Virtual DOS Machines
- Screen Savers

Startup Folder

Remove applications from the Startup folders of Windows NT Server console operators. For example, if MS Office Pro 95 is installed on the server, make sure that the FindFast indexing utility is not being loaded. The Office 95 Shortcut Toolbar is also unnecessary (using upwards of 3 MB when launched).
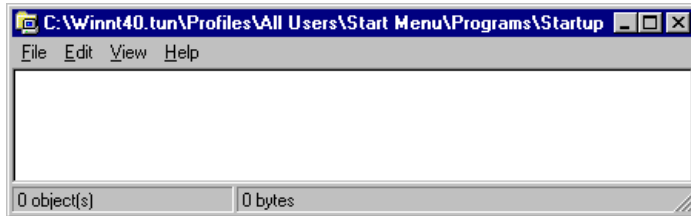
Virtual DOS Machines

Database administrators (DBAs) frequently run SQL scripts from MS-DOS command prompts (also known as virtual DOS machines, or VDMs). While VDMs are relatively lightweight, screen painting VDMs is expensive. When executing long-running scripts from a VDM, minimize the window so that the system can focus on the operation and not on a flood of window repaint messages.

Screen Savers

The most entertaining screen savers quickly saturate the CPU. If a screen saver must be run, choose Blank Screen, which uses the least amount of processing time.

To set a blank screen saver:

1. Choose Start > Settings > Control Panel > Display.

The *Display Properties* dialog box appears.

2. Click the Screen Saver tab.

The following information appears:
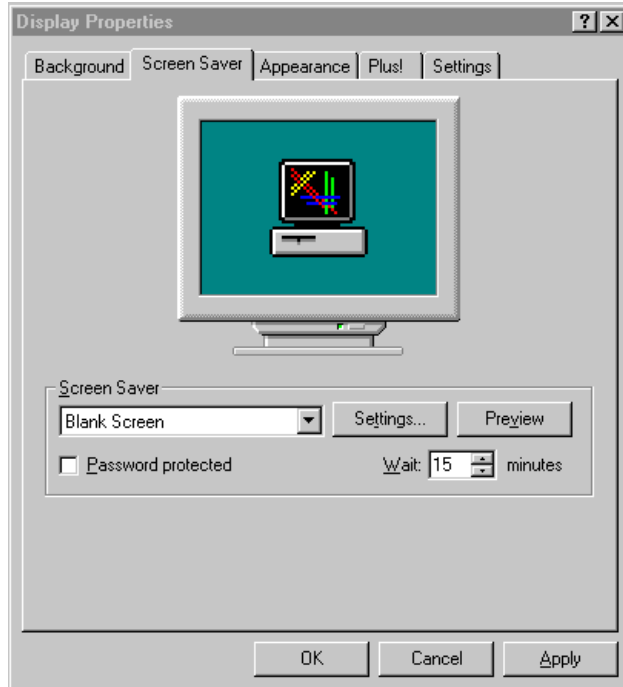
3. Select Blank Screen from the Screen Saver drop-down list box.

4. Click OK.

5. Exit the Control Panel.

11

Backing Up and Recovering Database Files

This chapter provides information on backing up and recovering your database. Specific topics discussed are:

- Selecting a Backup and Recovery Tool
- Backing Up Files with OCOPY
- Recovering Files with OCOPY
- Using Legato Storage Manager

Selecting a Backup and Recovery Tool

Backing up and recovering your Oracle8*i* database is one of the most critical operations that a database administrator (DBA) performs. For this reason, it is extremely important to choose the correct tools. The table below describes available backup and recovery tools and makes recommendations on which tool to use with your Oracle8*i* database:

Tool	Description	Analysis of Tool	To Use this Tool, See...
Oracle Enterprise Manager Backup Management tools	The Oracle Enterprise Manager DBA Management Pack includes the Backup Wizard for backing up and the Recovery Wizard for recovering your Oracle8*i* database. The Backup Wizard enables users to: ■ Perform database backups to disk and tape. ■ Create backup scripts. ■ Specify frequency, time, backup configuration, and multiple destinations. ■ Back up an entire database or tablespaces, data files, and archived redo logs. ■ Perform both hot and cold backups. The wizards provide a graphical user interface (GUI) to Recovery Manager.	Highly recommended for backing up and recovering your Oracle8*i* database because of capabilities and ease of use. Works in conjunction with Recovery Manager (RMAN). Backing up to tape requires an optional media management layer (MML) from a third-party vendor, such as Legato Storage Manager (LSM). LSM is included on your CD-ROM.	*Oracle Enterprise Manager Administrator's Guide*
Recovery Manager (RMAN) in command line mode[1]	Recovery Manager in command line mode is an Oracle8*i* tool that manages the process of backing up, restoring, and recovering files. Recovery Manager is automatically installed with Oracle Utilities. Recovery Manager uses a special PL/SQL interface to the server for invoking backup functions. The user is unaware of this interface and instead interfaces only with a command line.	Command line usage only. Must clearly understand the command line syntax described in *Oracle8i Recovery Manager User's Guide and Reference* before using. Backing up to tape requires an optional MML from a third-party vendor.	Chapter 2, "Database Tools Overview" for instructions on accessing this tool and *Oracle8i Recovery Manager User's Guide and Reference* for instructions on using this tool.

Tool	Description	Analysis of Tool	To Use this Tool, See...
Third-party vendor products	Third-party vendors such as Legato provide tape backup software that works with Backup Wizard and RMAN.	If using Legato MML, can back up to tape only if you are using RMAN and only if the tape device and the target database are on the same physical device. Contact your third-party vendors for additional information.	*Legato Storage Manager Administrator's Guide* and other third-party vendor documentation.
OCOPY	OCOPY is needed when performing a hot backup manually (that is, backing up a tablespace that is in offline backup mode).	Command line usage only. Can back up only to disk. Can only perform local backups.	"Backing Up Files with OCOPY" and "Recovering Files with OCOPY" in this chapter.

[1] On Windows NT, starting with version 8.1.6, additional transformations are added to file names entered by the user. If you use Recovery Manager, existing file names stored in the recovery catalog must be normalized. To normalize file names, see Chapter 8, "After Migrating or Upgrading the Database" in the *Oracle8i Migration* guide.

> **Note:** Before performing a cold backup, and regardless of the backup tool you use (for example, Oracle Enterprise Manager Backup Wizard or a third party tool), stop the OracleService*SID* service in order to clear locks on the database files. If you do not stop this service, some database files may not be backed up.

Backing Up Files with OCOPY

Use OCOPY for both hot and raw database file backups of the following file types:

File	Description
File Allocation Table (FAT) *or* NT File System (NTFS) files	File allocation table (FAT) or NT file system (NTFS) files you want to back up.
Logical raw files:	A logical drive (accessed with the direct I/O) identified by the name:
	`\\.\x:`
Note: For more information on logical drives, see Appendix D, "Storing Tablespaces on Raw Partitions"	where *x:* is the logical drive designator.
Physical raw files	A physical hard drive (accessed with direct I/O) identified by a device name of the form:
	`\\.\physicaldriveN`
	where *N* is 0, 1, 2, and so on, representing each of the physical drives in the system.

This table describes the two copy modes in which to use OCOPY:

Copy Mode	OCOPY is used to...
Hard disk	Copy hot files to a new name and location on a hard disk where you can use an archive utility to back them up.
Multiple diskettes	Back up large hot files directly to multiple diskettes.

This table describes the syntax to use when backing up all file types to hard disk or multiple diskettes:

Copy Mode	File Type	Syntax
Hard disk	FAT or NTFS files	`C:\> OCOPY old_file new_file`
	Logical raw	`C:\> OCOPY \\.\C: new_file`
	Physical raw	`C:\> OCOPY \\.\physicaldriveN new_file`
Multiple diskettes	FAT or NTFS files	`C:\> OCOPY /B hot_file a:`
	Logical raw	`C:\> OCOPY /B \\.\C: a:`
	Physical raw	`C:\> OCOPY /B \\.\physicaldriveN a:`

Where...	Indicates the...
old_file	Name and location of the hot file you want to back up.
new_file	Name and location of the backup copy.
C:	Raw drive that holds a single RAW database file.
physical drive	Physical drive that holds a single RAW database file.
N	Number(s) representing each of the physical drives in the system.
hot_file	Path and filename of the hot file you want to back up.
a:	Diskette drive containing the diskette on which to save the backup copy. If the file is too large to fit on one diskette, OCOPY prompts you to insert new diskettes as needed. OCOPY catalogs the parts of the file automatically so that it can be reconstructed at a later time.
/B	Large files must be split over multiple diskettes.

Note: Always use a fresh diskette for each OCOPY procedure; do not use OCOPY to back up a file onto a diskette that contains part or all of another file backed up using OCOPY.

Recovering Files with OCOPY

The table below describes the syntax to use when recovering all file types from hard disk or multiple diskettes.

Restore From...	File Type to Restore	Syntax
Hard Disk[1]	FAT or NTFS files	Use the Windows NT COPY command.
	Logical raw	`C:\> OCOPY new_file \\.\c:`
	Physical raw	`C:\> OCOPY new_file \\.\physicaldriveN`
Multiple diskettes[2]	FAT or NTFS files	`C:\> OCOPY /R a: restore_dir` **Note:** First insert the diskette containing the initial part of the backed up file.
	Logical raw	`C:\> OCOPY /R a: \\.\c:`
	Physical raw	`C:\> OCOPY /R a: \\.\physicaldriveN`

[1] These are files originally backed up without the /B option.
[2] These are files originally backed up with the /B option.

Where...	Indicates the...
C:	Raw drive, which holds a single RAW database file. If you use a different drive as your raw drive, substitute it for C:. OCOPY prompts you to insert diskettes as needed.
new_file	File name to which to restore the file.
physicaldrive	Physical drive that holds a single RAW database file.
N	Number(s) representing each of the physical drives in the system.
/R	Restore option.
a:	Drive containing the diskette with the backed up file.
restore_dir	Directory on the server in which to place the file. (The restored file has the same name as the original file.)

Using Legato Storage Manager

When running backups to tape drivers using **Legato Storage Manager** (LSM), you must add LSM's BIN directory to the PATH. The default location of the directory is C:\WIN32APP\NSR\BIN.

To add this directory to the PATH:

1. Select System from the Control Panel.

    The *System Properties* dialog box appears.

2. Select the Environment tab.

    The Environment view appears.

3. Select Path from the System Variables box.

4. Add the value, and click Set.

5. Click OK to close the *System Properties* dialog box.

Settings for Multiple Tape Devices

If you use LSM, and have more than one tape device on your system (LSM supports a maximum of 4), ensure that the following settings are made in the LSM Administrator GUI:

1. Select Start > Programs > NetWorker group > NetWorker Administrator to start the LSM Administrator GUI.

2. Click the set up server button, and set the parallelism field to the number of tape devices you have. Then, click the devices button. For each tape device listed, select it with the right mouse button, select edit, and set the target sessions field to 1.

Japanese Version of Legato Storage Manager

In the Japanese version of Legato Storage Manager 5.5, the following log files are written in UTF8 format:

C:\WIN32APP\NSR\APPLOGS\DMO.MESSAGES

C:\WIN32APP\NSR\LOGS\DAEMON.LOG

C:\WIN32APP\NSR\LOGS\MESSAGES

On a Japanese version of Windows NT running Japanese Legato Storage Manager 5.5, you must run the following command at the command prompt to read one of these UTF8 log files:

```
C:\> NSRCAT −N < FILE_NAME
```

where *FILE_NAME* is the appropriate path name for the specific log file, such as:

```
C:\WIN32APP\NSR\APPLOGS\DMO.MESSAGES
```

12

Developing Applications

This chapter describes topics of interest to application developers on Windows NT. Specific topics discussed are:

- Finding Information on Application Development for Windows NT
- Building External Routines
- Accessing Web Data with Intercartridge Exchange

Finding Information on Application Development for Windows NT

The following table describes where to find the information on developing applications specifically for Windows NT. These products are included on the CD-ROM.

To find information on...	Look in the guide...
Oracle JServer or Oracle JServer Enterprise Edition Oracle8*i* includes Oracle JServer, the integrated Java Virtual Machine, and provides Java2 support (JDK1.2), a CORBA 2.0 Object Request Broker, an embedded JDBC driver, a SQLJ translator, and an Enterprise JavaBeans transaction server.	■ *Supplied Java Packages Reference* ■ *Oracle8i Java Developer's Guide* ■ *Oracle8i Java Stored Procedures Developer's Guide*
XML Oracle's XML products include the XML Parser for Java - Version 1 and Version 2 (which includes an XSLT processor), the XML Class Generator, and the XML Parsers for C, C++, and PL/SQL as well as demos, utilities and sample code designed to illustrate the simplest and most powerful ways to work with XML-formatted data.	*Oracle8i Application Developer's Guide - XML*
Internet Tools Oracle WebDB enables you to publish your data to the Web	*Oracle WebDB Installation Guide and Tutorial* **Note:** WebDB is available on a separate CD-ROM and included with Oracle8*i* for Windows NT.
Application Wizards Oracle Application Wizards allow developers to create database applications easily and quickly. They improve ease-of-use and reduce development time by generating much of the code for database connectivity.	■ *Oracle AppWizard for Microsoft Visual C++ User's Guide for Windows NT* ■ *Oracle AppWizard for Microsoft Visual InterDev for Windows NT*
OLE Automation	■ *Oracle COM Automation Developer's Guide for Windows NT* ■ Oracle Objects for OLE (online Help) for Windows NT
Oracle Services for MTS Oracle8*i* for Windows NT provides Oracle Services for Microsoft Transaction Server (MTS). A Windows NT service called Oracle Service for MTS, permits enhanced deployment of COM components in MTS, using an Oracle database as the resource manager.	*Using Microsoft Transaction Server With Oracle8 for Windows NT*

To find information on...	Look in the guide...
Pro*C/C++ and Pro*COBOL—based applications	■ *Pro*C/C++ Precompiler Getting Started for Windows NT*
	■ *Pro*COBOL Precompiler Getting Started for Windows NT*
	■ *Oracle Call Interface Getting Started for Windows NT*
Writing external routines and the call specification	■ This chapter.
	■ *PL/SQL User's Guide and Reference*
	■ *Oracle8i Java Stored Procedures Developer's Guide*
	■ *Oracle8i Application Developer's Guide - Fundamentals*, Chapter 10 "External Routines."
	■ The following files in *ORACLE_BASE\ORACLE_HOME*\RDBMS\EXTPROC:
	EXTERN.C (code example shown in "Step 2: Writing an External Routine")
	MAKE.BAT (batch file that builds the dynamic link library)
	EXTERN.SQL (automates the instructions described in "Step 4: Registering an External Routine" and "Step 5: Executing an External Routine")
	README.DOC (explains how to run the sample and provides debugging advice)
OLE DB	*Oracle Provider for OLE DB*

Additional Information: Oracle ODBC Driver Release 8.1.6 is included on your CD-ROM. This driver is updated on a regular basis. To download the latest release of this driver, visit the following Web site:

```
http://technet.oracle.com/software/utilities/software_
index.htm
```

Building External Routines

This section describes how to create and use external routines on Windows NT.

External Routines Overview

External routines, previously referred to as external procedures, are functions written in a third-generation language (3GL), such as C, and callable from within PL/SQL or SQL as if they were a PL/SQL routine or function. External routines let you take advantage of the strengths and capabilities of a 3GL programming language in a PL/SQL environment.

> **Note:** Oracle also provides a special purpose interface, the call specification, that lets you call external routines from other languages, as long as it is callable by C.

The main advantages of external routines consist of the following:

- Performance, because some tasks are performed more efficiently in a 3GL language than in PL/SQL, which is better suited for SQL transaction processing
- Code re-usability, because dynamic link libraries (DLLs) can be called directly from PL/SQL programs on the server or in client tools such as SQL*Forms

You can use external routines to perform specific processes, such as the following:

- Solving scientific and engineering problems
- Analyzing data
- Controlling real-time devices and processes

Creating and using an external routine would involve the following sequential tasks:

- Step 1: Installing and Configuring
- Step 2: Writing an External Routine
- Step 3: Building a DLL
- Step 4: Registering an External Routine
- Step 5: Executing an External Routine

> **Attention:** You can combine the instructions described in the fourth and fifth tasks into one SQL script that automates the process of registering and executing your external routine. For an example of a SQL script that combines these steps, see *ORACLE_BASE\ORACLE_HOME*\RDBMS\EXTPROC\EXTERN.SQL.

Step 1: Installing and Configuring

This section describes the installation and configuration of the Oracle8*i* database and Net8.

Installing the Oracle8*i* database

Follow the steps in the *Oracle8i Installation Guide for Windows NT* to install these products on your Windows NT server:

- Oracle8*i* Enterprise Edition or Oracle8*i*. Contains PL/SQL, from which external routines are called, and the PL/SQL external routine program (EXTPROC), which executes external routines.
- Net8 Client, Net8 Server, and Oracle Protocol support

> **Note:** You must also have a C compiler and linker installed on your system to build DLLs.

Configuring Net8

If you install Net8 Server from your CD-ROM, your server network files are automatically configured to use external routines.

When PL/SQL calls an external routine, the Net8 listener launches a session-specific process called EXTPROC. Through Net8, PL/SQL passes the following information to EXTPROC:

- DLL name

- External routine name

- Parameters (if necessary)

EXTPROC then loads the DLL and invokes the external routine.

If you copy your Oracle7 server network files into your Oracle8*i* network files directory, you must manually configure the following files for the external routine behavior described previously to occur:

- *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN\LISTENER.ORA

- *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN\TNSNAMES.ORA

See Chapter 8 of the *Net8 Administrator's Guide* for instructions.

> **Note:** The SQLNET.ORA file requires no changes. By default, the values for the parameters NAMES.DEFAULT_DOMAIN and NAME.DEFAULT_ZONE are set to WORLD. These values match with the .WORLD extension on the end of EXTPROC_ CONNECTION_DATA in the TNSNAMES.ORA file.

Step 2: Writing an External Routine

Using a 3GL programming language, such as C, you can write functions to be built into DLLs and invoked by EXTPROC. The following is a simple Microsoft Visual C++ example of an external routine:

> **Note:** Since external routines are built into DLLs, they must be explicitly exported. In this example, the dllexport storage class modifier exports the function *find_max* from a dynamic link library.

```
#include <windows.h>
#define NullValue -1
/*
  This function simply returns the larger of x and y.
*/
long __declspec(dllexport) find_max(long x,
 short x_indicator,
long y,
short y_indicator,
```

```
      short *ret_indicator)
{
   /* It can be tricky to debug DLL's that are being called by a process
      that is spawned only when needed, as in this case.
      Therefore try using the DebugBreak(); command.
      This will start your debugger.  Uncomment the line with DebugBreak();
      in it and you can step right into your code.
   */
   /* DebugBreak();  */

   /* first check to see if you have any nulls */
   /* Just return a null if either x or y is null */

   if ( x_indicator==NullValue || y_indicator==NullValue) {
      *ret_indicator = NullValue;
      return(0);
   } else {
      *ret_indicator = 0;           /* Signify that return value is not null */
      if (x >= y) return x;
      else return y;
   }
}
```

Step 3: Building a DLL

After writing your external routine(s) in a 3GL programming language, use the appropriate compiler and linker to build a DLL, making sure to export the external routines, as noted above. See your compiler and linker documentation for instructions on building a DLL and exporting its functions.

After building the DLL, you can move it to any directory on your system. For the example above, you can build the external routine find_max into a DLL called EXTERN.DLL. To build the above example, go to *ORACLE_BASE\ORACLE_ HOME*\RDBMS\EXTPROC and type MAKE.

Step 4: Registering an External Routine

Once you have built a DLL containing your external routine(s), you must register your external routine(s) with the Oracle8*i* database:

1. Create a PL/SQL library to map to the DLL.

2. Start SQL*Plus:

```
C:\> SQLPLUS
```

3. Connect to the database with the appropriate user name and password.

4. Create the PL/SQL library using the CREATE LIBRARY command:

```
SQL> CREATE LIBRARY externProcedures AS 'C:\ORACLE\ORA81\RDBMS\
EXTPROC\EXTERN.DLL';
```

Where...	Represents the...
externProcedures	Alias library (essentially a schema object in the database)
C:\ORACLE\ORA81\RDBMS\EXTPROC\EXTERN.DLL	Path to the Windows NT operating system DLL EXTERN.DLL. This example uses C:\ORACLE as your Oracle base and \ORA81 as your Oracle home.

Note: The DBA must grant EXECUTE privileges on the PL/SQL library to users who want to call the library's external routine from PL/SQL or SQL.

5. Create a PL/SQL program unit specification.

Do this by writing a PL/SQL subprogram that uses the EXTERNAL clause instead of declarations and a BEGIN...END block. The EXTERNAL clause is the interface between PL/SQL and the external routine. The EXTERNAL clause identifies the following information about the external routine:

- Name
- DLL alias
- Programming language in which it was written
- Calling standard (defaults to C if omitted)

```
CREATE OR REPLACE FUNCTION PLS_MAX(
      x BINARY_INTEGER,
      y BINARY_INTEGER)
RETURN BINARY_INTEGER AS
      EXTERNAL LIBRARY externProcedures

      NAME "find_max"

      LANGUAGE C

      PARAMETERS (
        x long,              -- stores value of x
        x_INDICATOR short,  -- used to determine if x is a NULL value
        y long,              -- stores value of y
        y_INDICATOR short,  -- used to determine if y is a NULL value

      RETURN INDICATOR short ); -- need to pass pointer to return value's indicator
                               -- variable to determine if NULL.
                  -- This means that my  function will be defined as:
                                  - long max(long x, short x_indicator,
                                  - long y, short y_indicator, short * ret_indicator)
```

DLL alias. You need EXECUTE privileges for this library.

External routine to call. If enclosed in double quotes, it becomes case-sensitive.

Language in which the external routine was written.

Step 5: Executing an External Routine

To execute an external routine, you must call the PL/SQL program unit (that is, the alias for the external function) that registered the external routine. These calls can appear in any of the following:

- Anonymous blocks

- Standalone and packaged subprograms

- Methods of an object type

- Database triggers

- SQL statements (calls to packaged functions only)

In "Step 4: Registering an External Routine", the PL/SQL function PLS_MAX registered the external routine find_max. Follow the steps below to execute `find_max`:

1. Call the PL/SQL function PLS_MAX from a PL/SQL routine named UseIt:

```
CREATE OR REPLACE PROCEDURE UseIt AS
        a integer;
        b integer;
        c integer;
BEGIN
        a := 1;
        b := 2;
        c := PLS_MAX(a,b);
        dbms_output.put_line('The maximum of '||a||' and '||b||' is  '||c);
END;
```

2. Run the routine:

```
SQL> EXECUTE UseIt;
```

Accessing Web Data with Intercartridge Exchange

This section discusses the following topics:

- Configuring Intercartridge Exchange

- Using Intercartridge Exchange

- UTL_HTTP Exception Conditions

- Exception Conditions and Error Messages

- Troubleshooting

Configuring Intercartridge Exchange

You must add a parameter to the registry before using Intercartridge Exchange.

To configure Intercartridge Exchange:

1. Start the registry editor from the MS-DOS command prompt:

```
C:\> REGEDT32
```

The *Registry Editor* window appears.

2. Add HTTP_PROXY to the registry subkey of the Oracle home directory that you are using. The location of this parameter is determined by how many Oracle home directories are on your computer:

If you have...	Add HTTP_PROXY to...
One home directory	HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME0
Additional directories	HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\ HOME*ID*
	where *ID* is incremented for each additional Oracle home directory on your computer.

3. Choose Add Value from the Edit menu.

The *Add Value* dialog box appears.

4. Type HTTP_PROXY in the Value Name text box and REG_SZ in the Data Type text box.
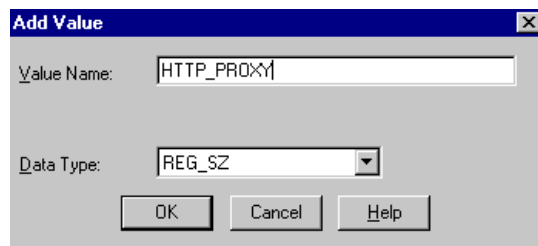
5. Click OK.

6. Type www-proxy.*your-site* in the String text box,
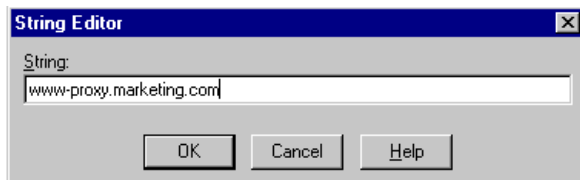
where *marketing.com* is an example of a Web site. (Type the domain name of your real Web site.)

Using Intercartridge Exchange

Intercartridge Exchange enables you to use a stored package called UTL_HTTP to make Hypertext Transfer Protocol (HTTP) calls from PL/SQL, SQL, and SQL*Plus statements.

UTL_HTTP can do both of the following:

- Access data on the Internet
- Call Oracle Web Application Server cartridges

UTL_HTTP contains two similar entry points, known as packaged functions, that turn PL/SQL and SQL statements into HTTP callouts:

- UTL_HTTP.REQUEST
- UTL_HTTP.REQUEST_PIECES

Both packaged functions peform the following tasks:

- Take a string universal resource locator (URL) of a site
- Contact that site
- Return the data (typically HTML) obtained from that site

The declarations to use with both packaged functions are described in the following subsections.

Packaged Function UTL_HTTP.REQUEST

UTL_HTTP.REQUEST uses a URL as its argument and returns up to the first 2000 bytes of data retrieved from that URL.

Specify UTL_HTTP.REQUEST as follows:

```
FUNCTION REQUEST (URL IN VARCHAR2) RETURN VARCHAR2;
```
To use UTL_HTTP.REQUEST from SQL*Plus, enter:

```
SQL> SELECT UTL_HTTP.REQUEST('HTTP://WWW.ORACLE.COM/') FROM DUAL;
```

which returns:

```
UTL_HTTP.REQUEST('HTTP://WWW.ORACLE.COM/')
----------------------------------------------------

<html>
<head><title>Oracle Corporation Home Page</title>
<!--changed Jan. 16, 19
1 row selected.
```

Packaged Function UTL_HTTP.REQUEST_PIECES

UTL_HTTP.REQUEST_PIECES uses a URL as its argument and returns a PL/SQL table of 2000 bytes of data retrieved from the given URL. The final element may be shorter than 2000 characters. The UTL_HTTP.REQUEST_PIECES return type is a PL/SQL table of type UTL_HTTP.HTML_PIECES.

UTL_HTTP.REQUEST_PIECES, which uses type UTL_HTTP.HTML_PIECES, is specified as:

```
type html_pieces is table of varchar2(2000) index by binary_integer;
function request_pieces (url in varchar2,
   max_pieces natural default 32767)
return html_pieces;
```

A call to REQUEST_PIECES can look like the example below. Note the use of the PL/SQL table method COUNT to discover the number of pieces returned; this may be zero or more:

```
declare pieces utl_http.html_pieces;
begin
  pieces := utl_http.request_pieces('http://www.oracle.com/');
   for i in 1 .. pieces.count loop
     .... -- process each piece
   end loop;
end;
```

The second argument to UTL_HTTP.REQUEST_PIECES, (MAX_PIECES) is optional. MAX_PIECES is the maximum number of pieces (each 2000 characters in length, except for the last, which may be shorter) that UTL_HTTP.REQUEST_ PIECES returns. If provided, that argument is usually a positive integer.

For example, the following block retrieves up to 100 pieces of data (each 2000 bytes, except perhaps the last) from the URL. The block prints the number of pieces retrieved and the total length, in bytes, of the data retrieved.

```
set serveroutput on
/
declare
  x utl_http.html_pieces;
begin
  x := utl_http.request_pieces('http://www.oracle.com/', 100);
  dbms_output.put_line(x.count || ' pieces were retrieved.');
  dbms_output.put_line('with total length ');
  if x.count < 1
  then dbms_output.put_line('0');
  else dbms_output.put_line
((2000 * (x.count - 1)) + length(x(x.count)));
  end if;
end;
/
```

which outputs:

```
Statement processed.
4 pieces were retrieved.
with total length
7687
```

The elements of the PL/SQL table returned by UTL_HTTP.REQUEST_PIECES are successive pieces of data obtained from the HTTP request to that URL.

UTL_HTTP Exception Conditions

This subsection describes the exceptions (errors) that can be raised by packaged functions UTL_HTTP.REQUEST and UTL_HTTP.REQUEST_PIECES.

UTL_HTTP.REQUEST

PRAGMA RESTRICT_REFERENCES enables the display of exceptions:

```
create or replace package utl_http is
function request (url in varchar2) return varchar2;
pragma restrict_references (request, wnds, rnds, wnps, rnps);
```

UTL_HTTP.REQUEST_PIECES

PRAGMA RESTRICT_REFERENCES enables exceptions to be displayed:

```
create or replace package utl_http is
type html_pieces is table of varchar2(2000) index by binary_integer;
function request_pieces (url in varchar2,
    max_pieces natural default 32767)
return html_pieces;
pragma restrict_references (request_pieces, wnds, rnds, wnps, rnps);
```

Exception Conditions and Error Messages

The following table describes error messages that may appear.

If...	Then...
Initialization of the HTTP callout subsystem fails for environmental reasons such as lack of available memory	Exception UTL_HTTP.INIT_FAILED is raised: `init_failed exception;`
The HTTP call fails due to failure of the HTTP daemon or because the argument to REQUEST or REQUEST_PIECES cannot be interpreted as a URL, because it is NULL or has non-HTTP syntax	Exception UTL_HTTP.REQUEST_FAILED is raised: `request_failed exception;`

If...	Then...

No response is received from a request to the given URL, because the function made no contact with a site corresponding to that URL	A formatted HTML error message may be returned:
	
	`<HTML>`
	`<HEAD>`
	`<TITLE>Error Message</TITLE>`
	`</HEAD>`
	`<BODY>`
	`<H1>Fatal Error 500</H1>`
	`Can't Access Document:`
	`http://home.nothing.comm.`
	`<P>`
	`<B>Reason:</B> Can't locate remote host:`
	`home.nothing.comm.`
	`<P>`
	`<P><HR>`
	`<ADDRESS><A HREF="http://www.w3.org">`
	`CERN-HTTPD3.0A</A></ADDRESS>`
	`</BODY>`
	`</HTML>`

> **Note:** The first two exceptions in the preceding table, unless explicitly caught by an exception handler, are reported by this generic message:
>
> ```
> ORA-06510: PL/SQL: unhandled user-defined
> exception
> ```
>
> that shows them as "user-defined" exceptions, although they are defined in this system package.
>
> If any other exception is raised during the processing of the HTTP request (for example, an out-of-memory error), then function UTL_HTTP.REQUEST or UTL_HTTP.REQUEST_PIECES reraises that exception.

Troubleshooting

Do not expect UTL_HTTP.REQUEST or UTL_HTTP.REQUEST_PIECES to succeed in contacting a URL unless you can contact that URL by using a browser on the same computer (and with the same privileges, environment variables, and so on).

If UTL_HTTP.REQUEST or UTL_HTTP.REQUEST_PIECES fails (that is, if it raises an exception or returns an HTML-formatted error message, yet you believe that the URL argument is correct), try contacting that same URL with a browser to verify network availability from your computer.

A

Directory Structures

This appendix describes the default directory structures created when you install Oracle8*i* components.

Specific topics discussed are:

- Oracle8i Directory Structure

- Oracle8i Client Directory Structure

- Filename Extensions

    See:

    - Chapter 3, "Multiple Oracle Homes and Optimal Flexible Architecture" for more information on Oracle homes and OFA. See especially "Directory Tree of a Sample OFA-Compliant Database" on page 3-19 for a depiction of how the default OFA-compliant directory tree is organized.

    - *Oracle Enterprise Manager Administrator's Guide* for directory structures for those products.

Oracle8*i* Directory Structure

ORACLE_BASE is the root of the Oracle directory tree.

Oracle Universal Installer places Oracle products into the *ORACLE_HOME*, \\ADMIN, and \\ORADATA directories of *ORACLE_BASE*.

The \\ADMIN and \\ORADATA directories contain one or more *DB_NAME* directories. *DB_NAME* is the unique name for a database and has the same value as the DB_NAME parameter in the INIT.ORA file. Database files are stored in *ORACLE_BASE*\\ORADATA*DB_NAME*.

> **See:** *Oracle8i Installation Guide for Windows NT* for the database files stored in *ORACLE_BASE*\\ORADATA*DB_NAME*.

The following sections describe these directories in *ORACLE_BASE*:

- ORACLE_HOME
- ADMIN

ORACLE_HOME

If you install Oracle8*i* Enterprise Edition or Oracle8*i* with the Typical installation type, the following directories are created. If you install additional Oracle products, other directories are also created.

Directory	Contents
\\ASSISTANTS	Oracle assistants. \\ASSISTANTS contains the following subdirectories:
\\DBCA	■ Oracle Database Configuration Assistant files
\\DBMA	■ Oracle Data Migration Assistant files
\\IFA	■ Oracle INTYPE File Assistant files
\\JLIB	■ Java files used by the assistants
\\OWAST	■ Oracle Web Publishing Assistant files
\\BIN	Executable files
\\CLASSES	Oracle Enterprise Manager files
\\COM	Oracle COM Automation feature files

Directory	Contents
\CTX	Oracle *inter*Media Text files. \CTX contains the following subdirectories:
\ADMIN	■ SQL scripts for installation and configuration, and header and body files for PL/SQL packages
\BIN	■ Executable files
\DEMO	■ Empty
\DOC	■ Release notes
\LIB	■ Library files
\MESG	■ Message files
\DATA	■ Data files
\LOG	■ Log file default directory
\MIGRATE	■ Migration scripts
\SAMPLE	■ Sample thesauri
\DATABASE	This is a legacy directory from previous releases. It contains an initialization parameter file that points to the new directory location for initialization parameter files.
\DBS	Oracle for Windows NT Performance Monitor files and other utility files
\DOC	HTML documentation library
\JAVAVM	Java Virtual Machine files.
\JDBC	Java Database Connectivity (JDBC) drivers files. \JDBC contains the following subdirectories:
\ADMIN	■ Class files
\DEMO	■ Sample programs
\JLIB	Java files used by various applications
LDAP	Directory Server files
\MD	Oracle Spatial files. \MD contains the following subdirectories:
\ADMIN	■ SQL scripts
\DOC	■ Release notes
\LIB	■ Library files

Directory	Contents
\MSHELP	Help files
\NETWORK	Net8 files. \NETWORK contains the following subdirectories:
\ADMIN	■ Configuration files
\AGENT	■ Oracle Intelligent Agent files
\DOCS	■ Release notes
\JLIB	■ Java files used by Net8 assistants
\LOG	■ Log files (default location)
\MESG	■ Message files
\TNSAPI	■ Net8 Open API-related files
\TOOLS	■ Net8 assistant files
\TRACE	■ Trace files (default location)
\OCI	Oracle Call Interface files. \OCI contains the following subdirectories:
\INCLUDE	■ Header files
\LIB	■ Library files
\SAMPLES	■ Sample files
\OCOMMON	NLS files
\ODBC	Oracle ODBC files
\OO4O	Oracle Objects for OLE files. \OO4O contains the following subdirectories:
\CPP	■ Header files, library files, and sample files
\EXCEL	■ Microsoft Excel sample files
\IIS	■ Internet Information Server sample files
\MESG	■ Message files
\VB	■ Visual Basic sample files
\OPSM*	Oracle Parallel Server files
\ORACORE	Message files
\ORD	Data option files. \ORD contains the following subdirectories:

Directory	Contents
\ADMIN	■ SQL scripts
\AUD	■ Oracle *inter*Media Audio files
\IM	■ Oracle *inter*Media files
\IMG	■ Oracle *inter*Media Image files
\MESG	■ Message files
\TS	■ Oracle Time Series files
\VID	■ Oracle interMedia Video files
\VIR*	■ Oracle Visual Information Retrieval files
\JLIB	■ Java libraries for Oracle options
\OTRACE	Oracle Trace files. \OTRACE contains the following subdirectories:
\ADMIN	■ Administration files, including SQL scripts
\DEMO	■ Sample programs
\LIB	■ Library files
\MESG	■ Message files
\PUBLIC	■ Header files
\OWM	Oracle Wallet Manager files*
\PLSQL	SQL scripts, sample files, and message files for PL/SQL
\PRECOMP	Precompiler files
\RDBMS	Oracle Server files. \RDBMS contains the following subdirectories:
\ADMIN	Oracle database SQL scripts (including CATALOG.SQL and CATPROC.SQL). Use SQL scripts to create data dictionary tables and views, and other views used by Oracle software. **Additional Information:** See Chapter 5, "SQL Scripts" in *Oracle8i Reference.*
\EXTPROC	External procedure sample files
\MESG	Message files

Directory	Contents
\XA	Oracle XA files
\SQLJ	Oracle SQLJ Translator files. \SQLJ contains the following subdirectories:
\DEMO	■ Sample programs
\DOC	■ Release notes, white papers, package descriptions, the SQLJ specification, and the *Oracle SQLJ Developer's Guide and Reference*
\LIB	■ Class files and a SQL script
\SQLPLUS	SQL*Plus files. \SQLPLUS contains the following subdirectories:
\ADMIN	■ SQL scripts for administering SQL*Plus
\DEMO	■ SQL scripts for sample tables
\DOC	■ Release notes
\MESG	■ Message files
\SVRMGL	Server Manager files
\SYSMAN	Oracle Enterprise Manager files

*Not included if you install Oracle8*i* for Windows NT.

ADMIN

Database administration files are stored in subdirectories of *ORACLE_BASE*\ADMIN*DB_NAME*.

The following table describes these subdirectories.

Directory	Contents
\ADHOC	Ad hoc SQL scripts
\BDUMP	Background process trace files (default location)
\CDUMP	Core dump files
\CREATE	Database creation files
\EXP	Database export files
\PFILE	Initialization parameter files
\UDUMP	User process trace files (default location)

Oracle8*i* Client Directory Structure

ORACLE_BASE is the root of the Oracle directory tree.

Oracle Universal Installer places Oracle8*i* Client products into the *ORACLE_HOME* directory of *ORACLE_BASE*.

If you install Oracle8*i* Client with the Programmer installation type, the following *ORACLE_HOME* directories are created.

Directory	Contents
\ASSISTANTS	Oracle assistants
\BIN	Executable files
\DOC	HTML documentation library
\JAVAVM	Java Virtual Machine files
\JDBC	Java Database Connectivity (JDBC) drivers files
\JLIB	Java files used by various applications
\LDAP	Files used by directory client libraries
\LIB	Java VM and EJB jar files
\MSHELP	Help files

Directory	Contents
\NETWORK	Net8 files
\OCI	Oracle Call Interface files. \OCI contains the following subdirectories:
\INCLUDE	■ Header files
\LIB	■ Library files
\SAMPLES	■ Sample files
\OCOMMON	NLS files
\ODBC	Oracle ODBC files
\OLEDB	OLE DB files
\OO4O	Oracle Objects for OLE files
\ORACORE	Message files
\ORD	Data option files. \ORD contains the following subdirectories:
\ADMIN	■ SQL scripts
\AUD	■ Oracle interMedia Audio files
\IMG	■ Oracle interMedia Image files
\MESG	■ Message files
\TS	■ Oracle Time Series files
\VID	■ Oracle interMedia Video files
\VIR*	■ Oracle Visual Information Retrieval files
\IM	■ Oracle interMedia files
\JLIB	■ Java libraries for Oracle options
\TRACE	Oracle Trace files
\OWM	Oracle Wallet Manager files
\PLSQL	Message files for PL/SQL
\PRECOMP	Precompiler files
\RDBMS	Oracle Server files
\RELNOTES	Release Notes
\SLAX	Message files

Directory	Contents
\SQLJ	SQLJ files
\SQLPLUS	SQL*Plus files

Filename Extensions

A description of filename extensions is shown below.

Extension	Description
.aud	Oracle audit file
.bmp	bitmap file
.c	C source file
.ctl	SQL*Loader control file; Oracle Server control file
.dat	SQL*Loader datafile
.dbf	Oracle Server tablespace file
.dmp	Export file
.doc	ASCII text file
.h	C header file; also, sr.h is a SQL*Report Writer help file
.jar	Java class archive
.lis	output of SQL*Plus scripts
.log	installation log files; Oracle Server redo log files
.mk	make files
.msb	NLS message file (binary)
.msg	NLS message file (text)
.o	object module
.ora	Oracle configuration files
.pc	Pro*C source file
.pco	Pro*COBOL source file
.sql	SQL* script files
.tab	SQL* script file
.trc	trace files

B

Oracle8*i* Database Specifications for Windows NT

Oracle8*i* uses initialization parameters on Windows NT to enable various features of the database every time an instance is started.

Specific topics discussed are:

- Initialization Parameter File (INIT.ORA) Overview
- Initialization Parameters Without Windows NT-Specific Values
- Calculating Database Limits

Initialization Parameter File (INIT.ORA) Overview

An initialization parameter file is an ASCII text file containing parameters. By changing the parameters and values in an initialization file, you can specify, for example:

- the amount of memory the database uses

- whether to archive filled online redo logs

- which control files currently exist for the database

Every database instance has a corresponding initialization parameter file and ORACLE_SID registry parameter that points to the system identifier (SID) for the instance.

The initialization parameter file name takes the form INIT.ORA. A single instance might have several initialization parameter files, each having some differences that affect system performance.

> **See:**
>
> - Your INIT.ORA file for initialization parameters set by Oracle Universal Installer during an Oracle8*i* typical installation type. These parameters may vary, depending on your hardware configuration.
>
> - *Oracle8i Reference* for descriptions of all initialization parameters and instructions for setting and displaying their values.

Location of the Initialization Parameter File

By default, Oracle8*i* uses the initialization parameter files located in *ORACLE_BASE*\ADMIN*DB_NAME*\PFILE, unless you specify a different initialization file with the PFILE option at database startup.

Editing the Initialization Parameter File

To customize Oracle8*i* database functions, you may need to edit the initialization parameter files. Only use an ASCII text editor to modify the file.

Sample File

A sample file called INITSMPL.ORA is located in the *ORACLE_BASE* \ADMIN\SAMPLE\PFILE directory.

> **Note:** If you create a database manually using the BUILD_DB.SQL script, you need to create an INIT.ORA file or copy an existing INIT.ORA file and modify the contents. If you use Oracle Database Configuration Assistant to create a database, the INIT.ORA file is automatically created for you.

If you want to use the sample INITSMPL.ORA file as part of database creation:

1. Rename the file INIT.ORA.

2. Edit this file to reflect the correct location of your database control files and the name of your database, as a minimum.

If you installed a starter database, the initialization parameter file INIT.ORA used by the starter database is located in *ORACLE_BASE*\ADMIN*DB_NAME*\PFILE. You can use either INITSMPL.ORA or the starter database INIT.ORA as a basis for creating a new Oracle8*i* database initialization parameter file.

The annotated, sample initialization parameter file contains alternative values for the initialization parameters. These values and the annotations are preceded by comment signs (#), which prevent them from being processed. To activate a particular parameter, remove the preceding # sign. When you no longer want to use a particular parameter, edit the initialization parameter file to add a comment sign.

For example, several initialization parameters are specified with three different values to create small, medium, or large System Global Areas (SGAs), respectively. The parameter that creates a small SGA is active in the following example:

```
  db_block_buffers = 200 # SMALL
# db_block_buffers = 550 # MEDIUM
# db_block_buffers = 3200 # LARGE
```

To create a medium-sized SGA, comment out the small parameter definition and activate the medium parameter definition. Edit the initialization parameter file as follows:

```
# db_block_buffers = 200 # SMALL
  db_block_buffers = 550 # MEDIUM
# db_block_buffers = 3200 # LARGE
```

Initialization Parameters Without Windows NT-Specific Values

Oracle8i Reference describes the default values for many initialization parameters as being *operating system-specific*. However, not all the parameters that *Oracle8i Reference* describes as having operating system-specific values affect Windows NT. In these cases, Windows NT uses either the default value set in the Oracle8i kernel or does not use the parameter. This table describes these initialization parameters:

Parameter	Description
AUDIT_FILE_DEST	Not supported on Windows NT and should not be added to the initialization parameter file.
DB_WRITER_PROCESSES	Not applicable or necessary on Windows NT.
COMPATIBLE_NO_RECOVERY	Uses default value set in Oracle8*i* kernel (no Windows NT-specific value).
CORE_DUMP_DEST	Not applicable to Windows NT.
CPU_COUNT	Oracle8*i* automatically sets value to number of CPUs available for your Oracle instance.
HI_SHARED_MEMORY_ADDRESS	Not applicable to Windows NT.
SHARED_MEMORY_ADDRESS	Not applicable to Windows NT.
LARGE_POOL_SIZE	Uses maximum value limited by available memory.
LOG_BUFFER	Starter database uses value set in Oracle8*i* kernel (no Windows NT-specific value). The Custom database creation option of the Oracle Database Configuration Assistant enables you to customize the value for this parameter.
ORACLE_TRACE_COLLECTION_PATH	Uses default value set in Oracle8*i* kernel (no Windows NT-specific value).
ORACLE_TRACE_FACILITY_NAME	Uses default value set in Oracle8*i* kernel (no Windows NT-specific value).
ORACLE_TRACE_FACILITY_PATH	Uses default value set in Oracle8*i* kernel (no Windows NT-specific value).
SPIN_COUNT	Uses default value set in Oracle8*i* kernel (no Windows NT-specific value).

Displaying Initialization Parameter Values

Windows NT-specific parameter values can be viewed by using an ASCII editor to open the *ORACLE_BASE*\ADMIN*DB_NAME*\PFILE\INIT.ORA file. To display *all* parameter values (whether set in the INIT.ORA file or the Oracle8*i* kernel), enter the following command at the SQL*Plus command prompt:

```
SQL> SHOW PARAMETER PARAMETER_NAME
```

where *PARAMETER_NAME* is the name of a specific initialization parameter.

The value for this parameter, whether defined in the *ORACLE_BASE*\ADMIN*DB_NAME*\PFILE\INIT.ORA file or the Oracle8*i* kernel, displays on-screen.

Database Initialization Parameters

Check the following initialization parameters when creating a new database. They *cannot* be modified after you have created the database. See Chapter 6, "Post-Installation Database Creation" for details on creating a new database, including the part of the procedure when you modify these parameters.

Parameter	Description
CHARACTER SET[1]	Specifies the database National Language Support (NLS) character set to use. This parameter can be set only when you create the database.
DB_BLOCK_SIZE	Specifies the size in bytes of Oracle database blocks.
DB_NAME	Specifies the name of the database to be created. The database name is a string of eight characters or less. You cannot change the name of a database.

[1] Not an initialization parameter, but rather a clause in the CREATE DATABASE statement. See Chapter 6, "Post-Installation Database Creation" for an example of using this clause.

Calculating Database Limits

Use the size guidelines in the following table to calculate Oracle8i database limits using the equations given in the *Oracle8i Administrator's Guide*.

Type	Size
Maximum block size	16,384 bytes or 16 kilobytes (KB)
Maximum blocks per file	4,194,304 blocks
Maximum possible file size with 16 K sized blocks	64 Gigabytes (GB) $(4,194,304 * 16,384) = 64$ gigabytes (GB)
Maximum number of files per database (depends on block size):	
2 K sized blocks	20,000 files
4 K sized blocks	40,000 files

Type	Size
8 K sized blocks	65,536 files
16 K sized blocks	65,536 files
Maximum file size for a FAT file	4 GB
Maximum file size in NTFS	16 Exabytes (EB)
Maximum database size	65,536 * 64 GB equals approximately 4 Petabytes (PB)
Maximum number of extents per database (depends on block size). Typical values are:	
2 KB sized blocks	121 extents
4 KB sized blocks	255 extents
8 KB sized blocks	504 extents
16 KB sized blocks	1032 extents
32 KB sized blocks	2070 extents
Shadow Process Memory:	
Release 8.1.6	335 K
Release 8.1.3	265 K
Release 8.0.4	254 K
Release 8.0.5	254 K

See: To calculate the space required by an index, use the equations given in *Oracle8i Administrator's Guide.*

C

Oracle8*i* Configuration Parameters and the Registry

This appendix describes use of the registry for various Oracle8*i* for Windows NT components. It also lists the recommended values and ranges for configuration parameters.

Specific topics discussed are:

- About Configuration Parameters

- Registry Overview

- Registry Parameters

- Oracle Parallel Server Registry Parameters

- Adding a Registry Parameter with REGEDT32

- Adding or Modifying Registry Parameters with Oracle Administration Assistant for Windows NT

- Modifying Oracle Performance Monitor for Windows NT Parameters

About Configuration Parameters

Oracle8*i* for Windows NT uses configuration parameters to locate files and specify runtime parameters common to all Oracle products. When an Oracle program or application requires a translation for a particular configuration variable, Oracle8*i* for Windows NT uses the associated parameter. All Oracle parameters are stored in the registry.

Registry Overview

Oracle8*i* for Windows NT stores its configuration information in a database (the registry) that is organized in a tree format. The tree format consists of keys in the registry and parameter values for the keys. Keys and parameter values can be viewed and modified in the Registry Editor.

Keys are folders that appear in the left pane of a Registry Editor window. A key contains subkeys or parameters.

> **WARNING:** Although the Registry Editor lets you view and modify registry keys and parameter values, you normally do not need to do so. In fact, you may render your system useless if you make incorrect changes. Therefore, only advanced users should edit the registry! Back up your system before making any changes in the registry.

Parameters in the Registry Editor appear as a string, consisting of three components:

- Parameter name
- Value class or type of entry
- Value itself

For example, parameter ORACLE_SID can have the following entry in the registry:

```
ORACLE_SID:REG_SZ:ORCL1
```

Value classes for Oracle8*i* for Windows NT parameters consist of the following:

- String value with a REG_SZ, REG_EXPAND_SZ (for an expandable string), or a REG_MULTI_SZ (for multiple strings) prefix to identify a parameter value entry as a data string

- Binary value with a REG_DWORD prefix to identify a value entry as a DWORD (hexadecimal data) entry

  Most Oracle8*i* for Windows NT parameter values are string types. Use Oracle Universal Installer defaults when a type is not given.

Registry Parameters

This section describes the Oracle8*i* for Windows NT registry parameters for the following keys. Other products, such as Oracle Enterprise Manager, have additional keys and parameters that are not described in this appendix.

- HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOMEID

- HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE

- HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\ALL_HOMES

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet

To modify the registry values described below, see "Modifying a Registry Value with REGEDT32" on page C-12.

Note: This appendix describes how to use REGEDT32 to edit your registry. If you are using Windows 95 or Windows 98, you must use REGEDIT. REGEDIT operates slightly differently than REGEDT32. See your Windows 95 or Windows 98 documentation for instructions.

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID*

Each time you install Oracle products into a new Oracle home on your computer, HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID* is created and *ID* is incremented. This subkey contains parameter values for most Oracle products.

> **Additional Information:** See Chapter 3, "Multiple Oracle Homes and Optimal Flexible Architecture" for details on the PATH variable and registry values when you are working with multiple Oracle homes.

This figure shows the parameter subkeys created with two Oracle home directories on the same computer.

The first Oracle installation on a computer creates parameter subkey HOME0.

The second Oracle installation on a computer creates parameter subkey HOME1. With each installation into a new Oracle home, HOME*ID* is created and incremented.
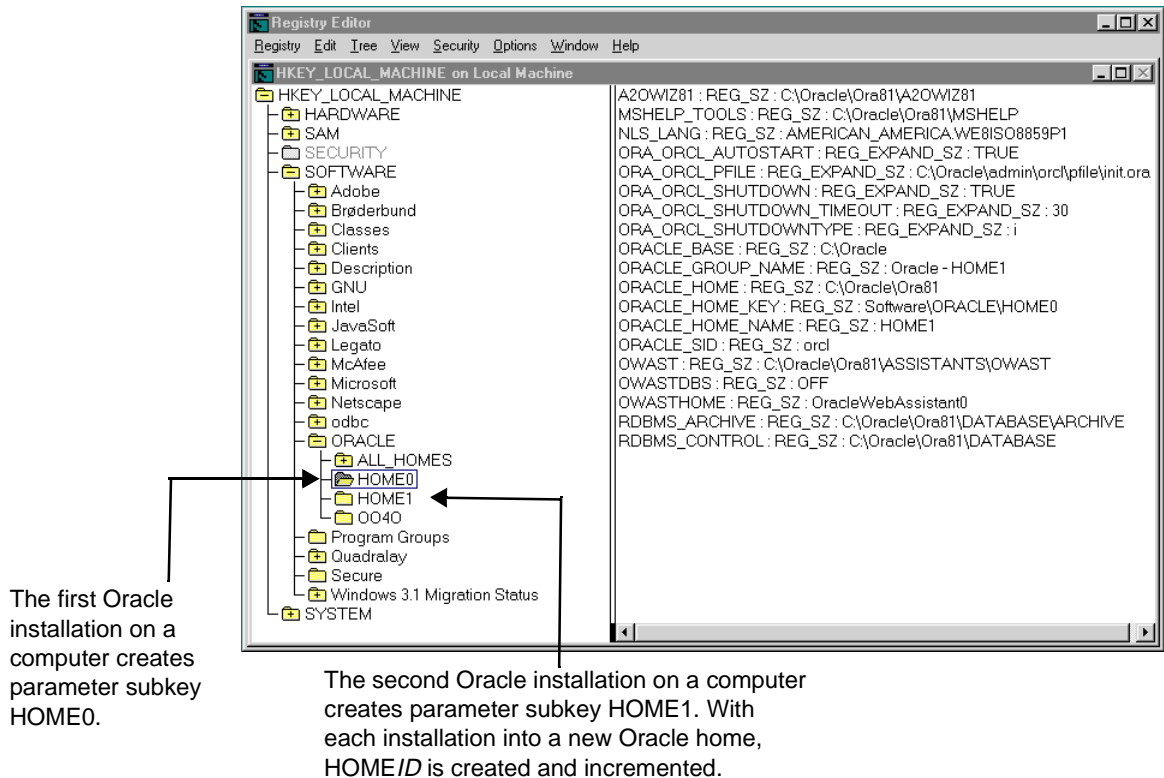
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID* includes the following parameters for an Oracle home directory on a computer. Depending on the products you install, additional parameters can also be created.

Parameter	Description	Default Value
MSHELP_TOOLS	Specifies the location of the Windows help files.	*ORACLE_BASE**ORACLE_HOME*\MSHELP
NLS_LANG	Specifies the supported language, territory, and character set. This parameter specifies the language in which the messages are displayed, the territory and its conventions for calculating week and day numbers, and the character set displayed.	During installation, Oracle Universal Installer sets this value based on the language setting of the operating system. See *Oracle8i Installation Guide for Windows NT* for a list of commonly used values.
		Note: If this parameter is deleted at a later time, Oracle uses the value AMERICAN_AMERICA.US7ASCII.
ORA_CWD	Specifies the current working directory. This parameter must be manually set. For example, if you set this parameter and then use ORADIM, a log file called ORADIM.LOG is created in this directory.	The value for this parameter must be set manually.
ORA_*SID_*AUTOSTART	Starts the database when the OracleService*SID* service is started.	TRUE
ORA_*SID_*PFILE	The full path to the initialization parameter file.	*ORACLE_BASE*\ADMIN*DB_NAME*\PFILE\INIT.ORA
ORA_*SID_*SHUTDOWN	When set to TRUE, shuts down the Oracle database identified by *SID* when OracleService*SID* is stopped.	TRUE
ORA_*SID_*SHUTDOWN_TIMEOUT	Sets the maximum time (in seconds) to wait for the shutdown to complete before the service for a particular SID stops.	30
ORA_*SID_*SHUTDOWNTYPE	The mode in which the database is shut down when you stop OracleService*SID*. The valid values are a (Abort), i (Immediate), and n (Normal).	i (Immediate)

Parameter	Description	Default Value
ORACLE_ AFFINITY	Specifies the Windows NT processor affinity of the threads within the Oracle process. The format is: `name1:cpumask1;name2:cpumask2` Each name setting must be the name of a background thread, USER for non-background (shadow) threads, and DEF for any thread type not handled specifically. The name MASK sets the affinity mask of the Oracle process. Valid background thread names are DBWR, LGWR, PMON, SMON, ARCH, RECO, CKPT, TRWR, SNP0 through SNP9, and P000 through P481. Each affinity setting must be a valid affinity mask (or its numeric equivalent) for the corresponding name. The process affinity mask is used only when the Oracle service is first started. Each thread's affinity is set only when the individual thread is started (for example, at database startup time for the background threads). **Note:** This parameter must be manually added.	The value for this parameter must be set manually. Oracle Corporation recommends consulting Oracle Support Services before changing this parameter.
ORACLE_BASE	The top-level Oracle directory (for example, C:\ORACLE) that contains *ORACLE_HOME*, \ADMIN, and \ORADATA.	*ORACLE_BASE*
ORACLE_GROUP_ NAME	Specifies the name of the group containing icons of the Oracle products installed. The parameter is added to your registry when you first install Oracle products, even if Oracle Universal Installer does not create a program group for the Oracle products you have installed (for example, if you have installed only Net8 software).	Oracle - *HOME_NAME*
ORACLE_HOME	Specifies the Oracle home directory in which Oracle products are installed. This directory is immediately beneath the Oracle base directory in the Oracle directory hierarchy.	The drive letter and name that you specify during installation
ORACLE_HOME_ KEY	The HKEY_LOCAL_MACHINE location of Oracle parameters.	SOFTWARE\ORACLE\HOME*ID*
ORACLE_HOME_ NAME	Specifies the home name of the Oracle home directory in which Oracle products are installed.	The name that you specify during installation

Parameter	Description	Default Value
ORACLE_ PRIORITY	Determines the Windows NT scheduling priorities of the threads within the Oracle ORDBMS or DBMS process. The format is: `name1:priority1;name2:priority2 . . .`	CLASS:normal; DEF:normal The name CLASS sets the priority class of the Oracle process. Threads can be assigned priority either collectively or individually. The collective name USER designates non-background (shadow) threads; the collective name DEF designates any thread type not handled specifically. Valid individual background thread names are DBWR, LGWR, PMON, SMON, ARCH, RECO, CKPT, TRWR, and SNP0 through SNP9. **Note:** ORACLE_PRIORITY is not automatically created for you in the registry. When it is not defined in the registry, the Windows NT default values are used for the priorities of the thread.
ORACLE_SID	Specifies the name of the Oracle database instance on the host machine. The value of this parameter is the SID for the instance.	The default value is specified by the entry in the *Database Identification* window of Oracle Universal Installer.
OWAST	Specifies the location of Oracle Web Publishing Assistant files.	*ORACLE_BASE\ORACLE_ HOME*ASSISTANTS\OWAST
OWASTDBS	Specifies whether database connection sharing is disabled.	OFF
OWAST_HOME	Specifies the name of the Oracle Web Publishing Assistant service.	OracleWebAssistant0
RDBMS_ ARCHIVE	Specifies the location of the backup database files.	*ORACLE_BASE\ORACLE_ HOME*DATABASE \ARCHIVE
RDBMS_ CONTROL	Specifies the location of the backup database control files.	*ORACLE_BASE\ORACLE_ HOME*DATABASE
SQLPATH	Specifies the location of SQL scripts.	*ORACLE_BASE\ORACLE_ HOME*DBS

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE

This subkey contains the following parameters:

Parameter	Description	Default Value Entry
INST_LOC	Specifies the location of Oracle Universal Installer files.	*System Drive:*\Program Files\Oracle\Inventory
OO4O	Specifies the location of Oracle Objects for OLE message files.	*ORACLE_BASE\ORACLE_HOME*\OO4O\MESG

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\ALL_HOMES

This subkey provides general information on each Oracle home directory on a computer. This subkey contains the ID*x* subkey(s) and its parameters, described below, as well as other parameters listed on page C-9.

ID*x*

This subkey corresponds to the HOME*ID* of the same number (for example, HOME0 for the first installation, HOME1 for the second installation, and so on). ID*x* contains the following parameters. The values that display are determined by what you enter during installation in the *File Locations* dialog box of Oracle Universal Installer.

Parameter	Description	Default Value Entry
NAME	Specifies the home name of the Oracle home for ID*x*. This is the value that you specify when prompted for an Oracle home name.	The name that you specify during installation.
PATH	Specifies the Oracle home directory for ID*x*.	*ORACLE_BASE\ORACLE_HOME*

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\ALL_HOMES Parameters

This subkey contains the following parameters.

Parameter	Description	Default Value
DEFAULT_HOME	Specifies the default Oracle home name (that is, the first Oracle home installed on your machine).	The name that you specify during installation.
HOME_COUNTER	Specifies the number of installed Oracle homes.	1
LAST_HOME	Displays the ID number of the most recently installed Oracle home. For example, if HOME0 was the most recently installed Oracle home, the number 0 appears.	0

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services

This subkey contains the following:

- Parameters for Oracle Performance Monitor for Windows NT

- Parameters for Oracle Services

Parameters for Oracle Performance Monitor for Windows NT

Oracle Performance Monitor for Windows NT parameters appear in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Oracle8\Performance.

> **Note:** Modify only the Hostname, Password, and Username values to point to any database.

For Oracle Performance Monitor for Windows NT to display information for Oracle performance objects, it must log onto the database. Modify the following parameters if the default information is not applicable or if you want to access another database:

Parameter	Description	Default Value
Hostname	Displays a Net8 connect string that allows you to edit the SID (two entries labeled *SID* in this example):[1]	Not applicable
	``` DESCRIPTION= (ADDRESS_LIST= (ADDRESS= (PROTOCOL=BEQ) (PROGRAM=oracle8) (ARGV0=oracle8SID) (ARGS='(DESCRIPTION=(LOCAL=YES)(ADDRESS= (PROTOCOL=beq)))'))) (CONNECT_DATA=(SID=SID))) ```	
Password	Displays the encrypted password for the user name to access the database.	MANAGER (encrypted)
Username	Displays the user name to access the database.	SYSTEM

[1] The 2:, 2:orcl, and 2:*sid* connect strings for local connections to Oracle8*i* for Windows NT are not supported. Instead, the default connect string for a local connection uses Bequeath Protocol support.

Oracle Performance Monitor for Windows NT requires the following parameters as entry points:

Parameter	Description	Default Value
Close	Specifies the close entry point for the DLL.	CloseOracle8PerformanceData
Collect	Specifies the collect entry point for the DLL.	CollectOracle8PerformanceData
Library	Specifies the name of the Oracle Performance Monitor DLL.	ORAPERF.DLL
Open	Specifies the open entry point for the DLL.	OpenOracle8PerformanceData

The following parameters specify the Oracle Performance Monitor for Windows NT log file and object configuration files:

Parameter	Description	Default Value
LOGFILE	Specifies the name of the Oracle Performance Monitor log file. This log file reports any errors, such as Oracle objects not displaying or database access problems.	*ORACLE_BASE\ORACLE_ HOME*\DBS \OPERF81.LOG
PERF_FILE_ NAME	Specifies the location of the PERF.ORA file. PERF.ORA contains all the performance objects displayed by Oracle Performance Monitor.	*ORACLE_BASE\ORACLE_ HOME*\DBS\PERF81.ORA

Parameters for Oracle Services

The HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services subkey contains additional subkeys that correspond to each Oracle service.

Each service subkey contains the following parameters:

Parameter	Description	Default Value Entry
DisplayName	Specifies the service name of the instance whose SID is *SID*.	Name of the service. For example, OracleServiceORCL1, where ORCL1 is the SID.
ImagePath	Specifies the fully qualified path name of the executable invoked by the service and any command-line arguments passed into the executable at runtime.	Path to the executable file of the product.
ObjectName	Specifies the logon user account and machine to which the service should log on.	LocalSystem

Oracle Parallel Server Registry Parameters

The following Oracle8*i* Parallel Server registry values are based on Oracle Corporation's reference implementation of the cluster Operating System Dependent (OSD) modules. Consequently, some of this information may not be applicable to your particular cluster environment. Consult with your hardware vendor for more details about installing and configuring your particular cluster configuration.

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD

This subkey contains the following required and optional values:

Value	Type	Description
CMDLL	REG_SZ	Specifies the full path of the Cluster Manager (CM) DLL. Required.
IODLL	REG_SZ	Specifies the full path of the Input/Output (IO) DLL. Required.
IPCDLL	REG_SZ	Specifies the full path of the Inter-Process Communication (IPC) DLL. Required.
PMDLL	REG_SZ	Specifies the full path of the Performance & Management (PM) DLL. PM is not an OSD component, but resides in this area. This DLL is maintained by Oracle. Required if using Oracle Enterprise Manager.
STARTDLL	REG_SZ	Specifies the full path of the Startup DLL. Optional.

Note: The Performance and Management (PM) DLL is no longer installed with the vendor OSD layer. The PM DLL is installed with Oracle8*i* Parallel Server in the *ORACLE_BASE\ORACLE_HOME*\BIN directory.

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\CM

This subkey contains the Cluster Manager (CM) registry values.

Value	Type	Description
ErrorLog	REG_SZ	Specifies the location of the CM error log file if CM is started by the Startup module. Oracle recommends using *SYSTEMROOT*\SYSTEM32\CM.LOG. If CM is started as a service, the error log is automatically placed in this location. In this case, it is not necessary to set a value for ErrorLog.
CmSrvrPath	REG_SZ	Specifies the full path of the CM executable. Required if the START module is used and CM is started by the Startup module.
DefinedNodes	REG_MULTI_SZ	A list of TCP/IP DNS host names. Each host name defines a potential member of the cluster. These host names must be the same as the host name in the *SYSTEMROOT*\SYSTEM32\DRIVERS\ETC\HOSTS file. Required. **Note:** The order of HostNames in this value is important since it defines the node numbers (0...*n*). Each host that is a cluster member must have exactly the same value.
PollInterval	REG_DWORD	Defines the check-in time among CMs on different nodes. Each CM is expected to send at least one status packet to all other nodes per poll interval. The default is 1000 milliseconds, or 1 second
MissCount	REG_DWORD	Defines the number of check-in intervals that can be missed before a CM and its related node are declared down by the cluster. The default is 3.
CmHostName	REG_SZ	Used by the local CM. Useful in a multihost environment, when more than one network is available on the node.
CmDiskFile	REG_SZ	Enables the Split-Brain detection feature. To enable, set this to a partition of size greater than 2 MB, for example: \\.\OPS_ CMDISK. Null value means the feature is disabled.

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\PM

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD*DB_NAME*

This subkey contains the Performance and Management (PM) registry values. Under this subkey, each Oracle8*i* Parallel Server cluster has its own registry subkey. The domain key name is the same as the cluster name. Within these keys, values are defined to provide the per-domain and per-instance information.

Value	Type	Description
0	REG_MULTI_SZ	Specifies the cluster instance ID data assigned to the first instance (OP1) on the primary node with the following format: *SID COMPUTER_NAME HOST_NAME ORACLE_HOME* For example: op1 OPSHIP1 opship1 c:\oracle\ora81
1	REG_MULTI_SZ	Specifies the cluster instance ID data assigned to the second instance (OP2) on the second node with the following format: *SID COMPUTER_NAME HOST_NAME ORACLE_HOME* For example: op2 OPSHIP2 opship2 c:\oracle\ora81
n	REG_MULTI_SZ	Specifies the cluster instance ID data assigned to the *n*th instance (OP*n*) on the *n*th node.

Modifying a Registry Value with REGEDT32

> **CAUTION:** Do not edit your registry unless absolutely necessary. If an error occurs in your registry, Oracle8*i* for Windows NT can stop functioning and the registry itself can become unusable.

To edit the Oracle-related settings:

1. Start the registry in one of two ways:

   - From the command prompt, enter:

     ```
     C:\> REGEDT32
     ```

   - Choose Start > Run, enter REGEDT32 in the Open field, and click OK.

   > **Note:** Use REGEDIT to edit the registry on Windows 95 and Windows 98. The dialog boxes for adding a registry parameter using REGEDIT are slightly different than those described below for REGEDT32. See your Windows 95 and Windows 98 documentation for specific instructions.

The *Registry Editor* window appears.
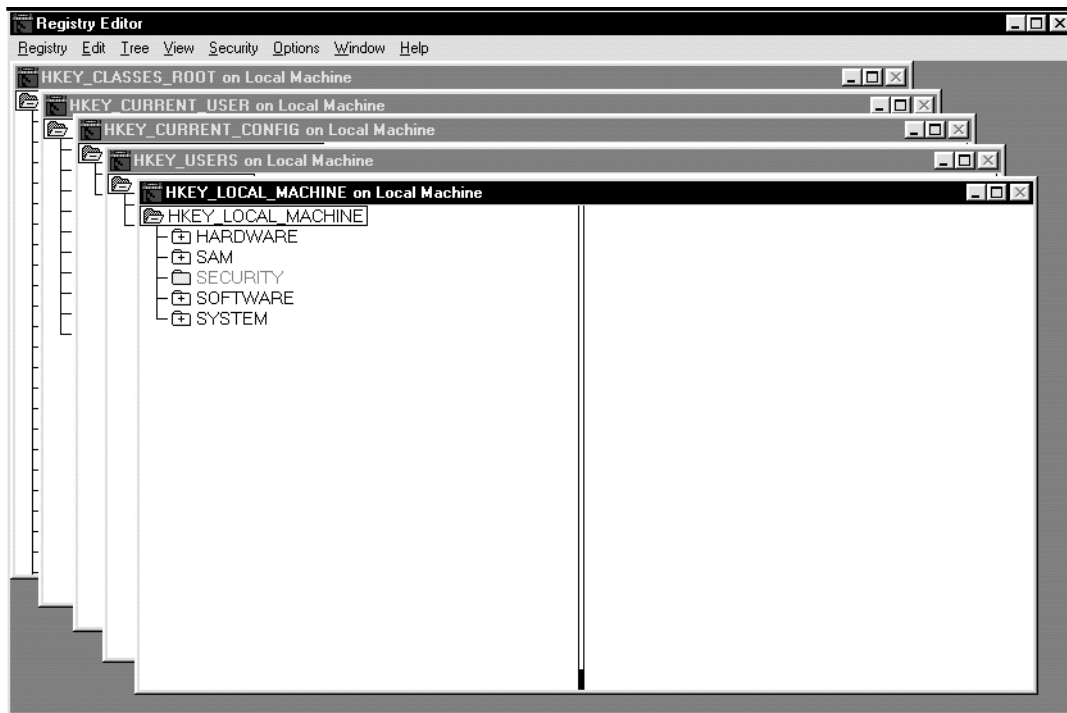
2. Navigate to the values you want to view or modify by double-clicking the appropriate keys.

   The left-hand side of the window shows the hierarchy of registry keys, and the right-hand side of the window shows various values associated with a key.

3. Double-click the parameter to edit.

   The *String Editor* dialog box appears:

4. Make any necessary edits.

5. Click OK.

6. Choose Exit from the Registry menu.

Adding a Registry Parameter with REGEDT32

To add a parameter to the registry:

1. Start the registry in one of two ways:

    - From the MS-DOS command prompt, enter:

      ```
      C:\> REGEDT32
      ```

    - Choose Start > Run, enter REGEDT32 in the Open field, and click OK.

      Note: Use REGEDIT to edit the registry on Windows 95 and Windows 98. The dialog boxes for adding a registry parameter using REGEDIT are slightly different than those described below for REGEDT32. See your Windows 95 and Windows 98 documentation for specific instructions.

    The *Registry Editor* window appears.

2. Navigate to the key to which you want to add the new value.

3. Choose Add Value from the Edit menu.

    The *Add Value* dialog box appears:
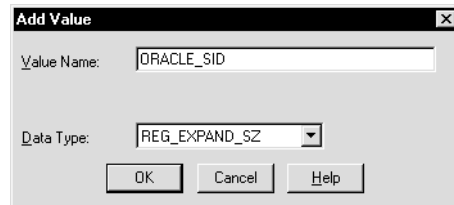
4. In the *Value Name* text box, type the name that you want to assign to the currently selected key.

5. In the *Data Type* list, select the value class that you want to assign to the added value:

- REG_SZ, REG_EXPAND_SZ (for an expandable string), or REG_MULTI_SZ (for multiple strings) for a data string

- Binary value with a REG_DWORD prefix to identify a value entry as a DWORD (hexadecimal data) entry

6. Click OK.

A *String Editor* dialog box appropriate for the data type appears:
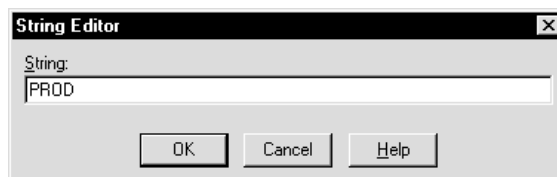
7. Type the value for the parameter.

8. Click OK.

The Registry Editor adds the parameter.

9. Choose Exit from the Registry menu.

The registry exits.

Adding or Modifying Registry Parameters with Oracle Administration Assistant for Windows NT

Instead of using REGEDT32 to add, edit, and delete parameters for an Oracle home, you can use the Oracle Home Configuration snap-in included as part of Oracle Administration Assistant for Windows NT. You must have Microsoft Management Console on your computer to use this product. The Oracle home parameters are located in the HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID* key.

See "HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOMEID" on page C-4 for more information about the Oracle home parameters.

Starting the Oracle Administration Assistant for Windows NT

To start the Oracle Administration Assistant for Windows NT:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Administration Assistant for Windows NT.

   Oracle Administration Assistant for Windows NT starts.

2. Expand Oracle Homes.

3. Right-click the Oracle home that you want to modify.

4. Click Properties.

   The *Properties* dialog box appears.

Adding Oracle Home Parameters

You can only add parameters with a data type of REG_SZ. Use REGEDT32 to add parameters with a data type of REG_EXPAND_SZ, REG_MULTI_SZ, or REG_DWORD.

To add an Oracle home parameter:

1. Click Add in the *Properties* dialog box.

2. Enter the name in the Parameter Name text box.

3. Enter the value in the Parameter Value text box.

4. Click OK.

5. Click Apply.

Editing Oracle Home Parameters

To change the default SID:

- Select the SID from the Default SID list in the *Properties* dialog box.

To edit one of the other parameters:

1. Select the parameter in the Other Settings text box in the *Properties* dialog box.

2. Click Edit.

3. Modify the value.

4. Click OK.

5. Click Apply.

Deleting Oracle Home Parameters

To delete an Oracle home parameter:

1. Select the parameter in the Other Settings text box in the *Properties* dialog box.

2. Click Delete.

Modifying Oracle Performance Monitor for Windows NT Parameters

Instead of using REGEDT32 to modify Oracle Performance Monitor's Hostname, Password, and Username parameters, you can use either of the tools described in this section.

See "Parameters for Oracle Performance Monitor for Windows NT" on page C-9 for more information about the Hostname, Password, and Username parameters.

Using OPERFCFG

OPERFCFG is an Oracle tool that you run from the MS-DOS command prompt.

The table below describes OPERFCFG command line syntax and provides examples of its use.

The -U, -P, and -D commands are all optional.

Use this syntax:	`OPERFCFG [-U USERNAME] [-P PASSWORD] [-D DATABASE_NAME]`	
Syntax description:	■ `-U USERNAME`	Username registry parameter value that Oracle Performance Monitor uses to log into the database. You must have DBA privileges on this database.
	■ `-P PASSWORD`	Password registry parameter value for the user name.
	■ `-D DATABASE_NAME`	Net service name that Oracle Performance Monitor uses to connect to the database. Affects the Hostname registry parameter. The net service name corresponds to the SID of the database that you want to monitor. The -D command can be specified without providing a database name value.
Example 1	`C:\> OPERFCFG -U DBA_ADMIN -P FRANK`	
Changing the Username and Password	This changes the user name to DBA_ADMIN and the password to FRANK and leaves the database name at its current value.	
Example 2	`C:\> OPERFCFG -U DBA_ADMIN -P FRANK -D PROD`	
Changing the Username, Password, and Database Name	This changes the username to DBA_ADMIN, the password to FRANK, and the database name to PROD.	
Example 3	`C:\> OPERFCFG -P FRANK`	
Changing the Password	This changes the password to FRANK for the current user name and database name.	
Example 4	`C:\> OPERFCFG -D`	
Changing the Database Name to an Empty Value	This changes the Hostname parameter to a blank value. This causes the Oracle Performance Monitor to connect to the default database on the computer. The current user name and password must be valid user accounts on this database.	

Using the Oracle Administration Assistant for Windows NT

The Oracle Performance Monitoring snap-in is part of the Oracle Administration Assistant for Windows NT. You must have Microsoft Management Console on your computer in order to use this product.

To use the Oracle Performance Monitoring snap-in:

1. Choose Start > Programs > Oracle - *HOME_NAME* > Database Administration > Oracle Administration Assistant for Windows NT.

   Oracle Administration Assistant for Windows NT starts.

2. Right-click Performance Monitor.

3. Click Properties.

   The *Performance Monitor Properties* dialog box appears.

4. Modify the text in the Username, Password, or Database text boxes and click Apply.

D

Storing Tablespaces on Raw Partitions

This appendix describes how to configure your system to store data files for tablespaces on raw partitions.

Specific topics discussed are:

- Raw Partition Overview
- Creating an Extended Partition
- Creating a Tablespace in a Raw Partition
- CRLOGDR Utility

> **Note:** Oracle Parallel Server requires additional configuration tools. See the *Oracle Parallel Server Administrator's Guide* on Windows NT for information on how to create logical partitions and assign symbolic links. Do not use this appendix to create partitions for Oracle Parallel Server.

Raw Partition Overview

In addition to storing data files for tablespaces on a file system, data files can also be stored on raw partitions.

A *raw partition* is a portion of a physical disk that is accessed at the lowest possible level. Input/output (I/O) to a raw partition offers approximately a 5% to 10% performance improvement over I/O to a partition with a file system on it.

A raw partition is created after generation of an extended partition and a logical partition, after partitions are assigned to i, and before applying any formatting. The Windows NT Disk Administrator application enables you to create an extended partition on a physical drive.

An extended partition points to raw space on the disk that can be assigned multiple logical partitions for the database files. An extended partition also avoids the four-partition limit by letting you define large numbers of logical partitions to accommodate applications using the Oracle8*i* database. Logical partitions can then be given symbolic link names to free up drive letters.

The Disk Administrator window below shows four disks. Two of the disks have an extended partition.
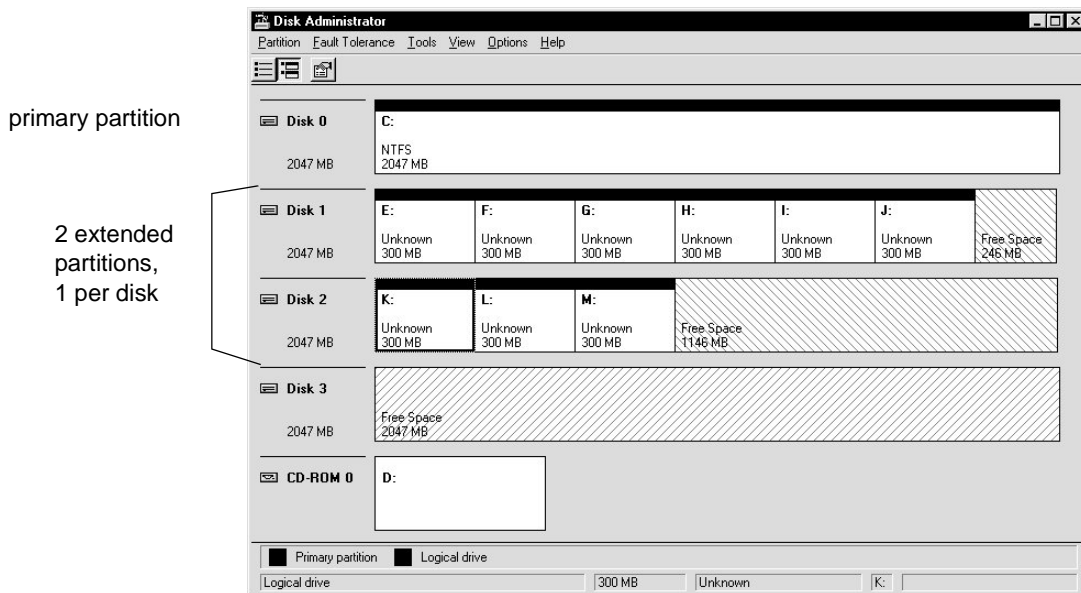
Disk No.	Contents
Disk 0	A primary partition
Disk 1	An extended partition with six logical partitions and 246 MB of free space
Disk 2	An extended partition with three logical partitions and 1146 MB of free space
Disk 3	An unformatted partition

Note: You can tell whether a partition is formatted or unformatted by the direction of the diagonal lines. A formatted partition's lines display from top left to bottom ($\backslash\backslash$), and an uninitialized partition's lines display from top right to bottom ($//$).

Disk Definition

Windows NT defines each disk drive found at startup with the following naming convention:

`\Device\Harddiskm\Partitionn`

where Harddisk*m* is the number of the physical drive, and Partition*n* is a logical partition number, as shown in the *Disk Administrator* window (in the above figure). Harddisk*m* starts at 0, and Partition*n* starts at 1.

Partition0 has a special meaning in that it has access to the whole disk. For example, the first logical partition (E:) on the second physical drive in the above figure has the following entry:

`\Device\Harddisk1\Partition1`

The first logical partition on a system (normally the C drive) has the following entry:

`\Device\Harddisk0\Partition1`

Raw Partition Definition

Raw partitions are of two types:

- Physical Disk
- Logical Partition

> **Note:** Although you can use physical disks, Oracle Corporation
> recommends that you use logical partitions.

Physical Disk

A physical disk represents the entire disk and points to

```
\Device\Harddiskx\Partition0.
```

Windows NT automatically creates a symbolic link name of \\.\PhysicalDrive*x*,
where *x* is the number corresponding to your hard disk drive number in the Disk
Administrator. The *x* matches the *x* in \Device\Harddisk*x*\Partition0.

\\.\PhysicalDrive*x* is automatically defined by Windows NT for every hard disk in
the computer. For example, a computer with three hard disks:

```
\\.\PhysicalDrive0
\\.\PhysicalDrive1
\\.\PhysicalDrive2
```

Internally, these names expand to

```
\\.\PhysicalDrive0 = \Device\Harddisk0\Partition0
\\.\PhysicalDrive1 =\Device\Harddisk1\Partition0
\\.\PhysicalDrive2 =\Device\Harddisk2\Partition0
```

Partition0 is special, because it represents the entire physical disk regardless of any
partitioning scheme on that disk. On all disks recognized by Windows NT, the Disk
Administrator writes a signature on the first block of all disks. To avoid overwriting
that block, Oracle skips the first block of a physical raw partition that is used for an
Oracle data file.

Logical Partition

A logical partition is a partition created by the Disk Administrator that points to a drive other than \Device\Harddisk*x*\Partition0.

Logical partitions are initially assigned names with drive letters (\\.*DRIVE_ LETTER*:) and typically re-assigned symbolic link names (\\.*SYMBOLIC LINK NAME*). For example, \\.\D: may be assigned a symbolic link name of \\.\ACCOUNTING_1. Regardless of whether a drive letter or symbolic link name is used, logical partitions are defined to represent a specific partition in a disk rather than the entire disk. Internally, these names may expand to:

```
\\.\D:= \Device\Harddisk2\Partition1
\\.\ACCOUNTING_1= \Device\Harddisk3\Partition2
```

Drive letters can be assigned to specific partitions, using the Disk Administrator. Symbolic link names can, on the other hand, be assigned using a utility such as DOSDEV.EXE, which is available with the Windows NT Resource Kit, or the SETLINKS utility.

> **Note:** Oracle does *not* skip the first block of a logical raw partition used for an Oracle data file.

Physical Disk and Logical Partition Considerations

Consider the following when deciding which raw partition to use:

- Physical disks are automatically defined by Windows NT to represent the entire disk, and should *never* be defined by the user.

- Logical partitions need to be defined by the user to represent a specific partition in a disk. These partitions should be logical partitions or drives contained in an extended partition. They should *never* be defined as Partition0.

- Using an entire disk (Partition 0) for an Oracle data file and using a partition that occupies the entire disk for an Oracle data file is *not* the same thing. Even when a partition occupies the entire disk, there is still a small space on the disk that is not part of the partition.

- If you are using an entire disk for an Oracle data file (Partition 0), use the pre-defined physical raw names that Windows NT provides.

- Use a logical partition if you are using a specific partition and it occupies the entire disk.

- If using a specific partition created with the Disk Administrator, define and use a symbolic link name rather than a logical partition number (even if it occupies the entire disk).

Frequently Asked Questions

Question: *What is the impact if I have created logical partitions, but defined physical disk convention names for them. For example:*

```
\\.\PhysicalDriveACCOUNTING_1 = \Device\Harddisk2\Partition1
\\.\PhysicalDriveACCOUNTING_2 = \Device\Harddisk3\Partition1
```

Answer: The Oracle database handles the data file using the physical disk convention even though it really is a logical partition. This will not cause any data corruption or loss as long as you continue to use the physical disk naming conventions. Oracle Corporation recommends that you convert to the logical partition at your earliest convenience. See "Compatibility Issues" on page D-7.

Question: *What is the impact if I have created logical names representing Partition0? For example:*

```
\\.\ACCOUNTING_1 = \Device\Harddisk1\Partition0
```

Answer: This poses severe problems, because the Disk Administrator typically writes a signature into the first block of every disk, and consequently may overwrite a portion of the data file header.

This can also cause data loss. Never use Partition0 with the logical partition convention. See "Compatibility Issues" on page D-7 for information on rebuilding your Oracle database with the proper conventions.

Question: *How do I transfer the contents of any raw partition to a standard file system for backup purposes?*

Answer: Use the Oracle-provided OCOPY tool to copy data to/from a raw partition for both physical and logical raw conventions. See "Compatibility Issues" on page D-7 below for further information.

Compatibility Issues

The physical and logical partition conventions are not compatible with one another because of the extra block that is skipped for physical raw conventions. This also means you cannot simply use OCOPY to copy from a physical disk to a logical partition, because the contents of these partitions are incompatible.

To convert from a physical convention to a logical convention, you must:

1. Perform a full database export to a (local) file system.

2. Create logical partitions and define logical names for these partitions.

3. Re-create the database by using the new logical partitions.

4. Perform the full database import to the newly-created database.

If your database installation uses physical disk conventions with logical partitions, Oracle Corporation recommends converting to the logical partition conventions at your earliest convenience, using the preceding steps.

Creating an Extended Partition

Only one extended partition can be created per disk. You can use the free space in the extended partition to create multiple logical partitions or use all or part of it when creating volume sets or other kinds of volumes for fault-tolerance purposes.

To create an extended partition:

1. Choose Start > Programs > Administrative Tools > Disk Administrator.

   The *Disk Administrator* window appears.
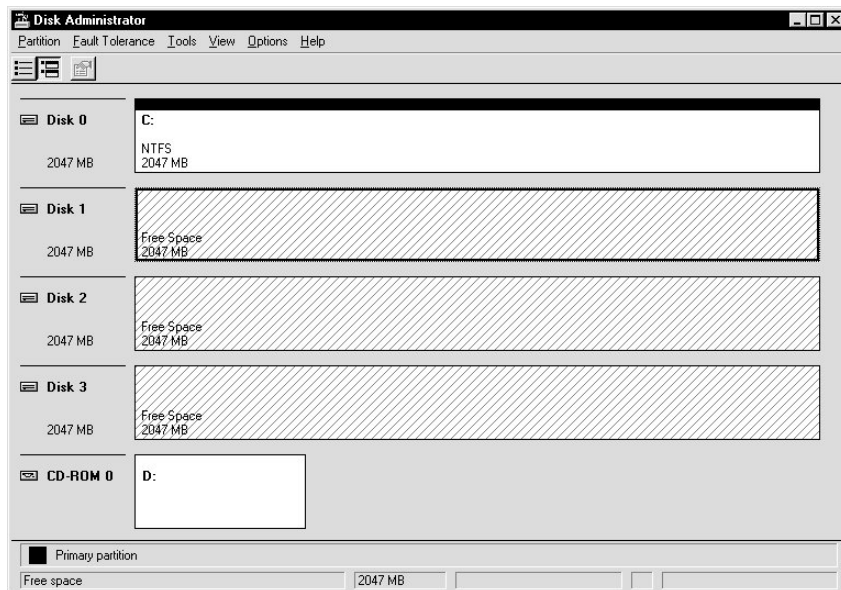
   Note that the lines display diagonally from top right to bottom left, indicating unpartitioned devices.

2. Select an area of free space in an extended partition on a disk that is on the shared disk subsystem by clicking the mouse.

   Oracle Corporation recommends that you use the entire disk.

3. Choose Partition > Create Extended.

Disk Administrator displays the minimum and maximum sizes for the
extended partition:
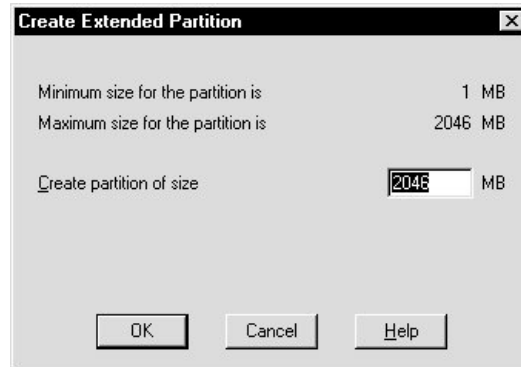
4. Use the default maximum size, then click OK.

> **Note:** Changes that you have made are not saved until you choose
> Partition > Commit Changes Now or exit Disk Administrator.

The extended partition is created.

Note that the lines now display diagonally from top left to bottom right,
indicating the partition is an extended partition.

Creating Logical Partitions in an Extended Partition

After an extended drive is created, you must assign logical partitions to it. Logical
partitions are assigned letters of the alphabet.

To create logical partitions in an extended partition:

> **Note:** Oracle Corporation recommends you do not create more
> than 120 logical partitions in an extended partition.

1. Select an area of free space in an extended partition by clicking the mouse on it.

2. Choose Partition > Create.

The *Disk Administrator* window displays the minimum and maximum sizes for the logical partition:
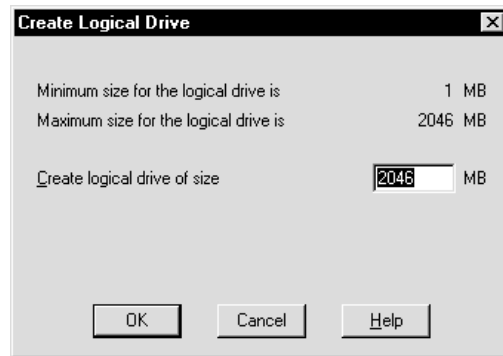
3. Enter the size of the logical partition for the data file, then click OK.

The size depends on how large you want your data files to be. Add 2 MB to this size for overhead.

4. Repeat Steps 1-3 for each additional data file that you plan to store in a raw partition.

5. Choose Partition > Commit Changes Now.

A confirmation dialog appears, informing you that changes have been made to the disk.

6. Click Yes.

A dialog box appears, informing you that the disks have been updated successfully.

7. Click OK.

8. Write down the hard disk number(s) and the number of the partition (starting at 1) for that drive. Oracle Corporation recommends using a worksheet similar to the one below.

Hard Disk Number	Partition Number Range
Hard Disk*x*	Partitions *x*-*x*
Hard Disk*x*	Partitions *x*-*x*

9. Choose Partition > Close.

Disk Administrator exits.

Assigning Symbolic Links to Each Logical Partition

If you do not want to use the drive letters assigned to the logical partitions that you created in "Creating Logical Partitions in an Extended Partition" on page D-9, use the Oracle SETLINKS utility to create symbolic links to raw logical partitions. Otherwise, keep the drive letters and proceed to "Creating a Tablespace in a Raw Partition" on page D-13.

The SETLINKS utility requires an input ASCII file that maps a symbolic link name to a raw partition. The following figure shows an ASCII file that maps two symbolic links, ACCOUNTING_1 and ACCOUNTING_2, to two logical partitions on the first disk:

```
                           Physical drive
                           number
        Symbolic link name                   Partition number



ACCOUNTING_1 \Device\Harddisk1\Partition1
ACCOUNTING_2 \Device\Harddisk1\Partition2
```

To assign symbolic links to each logical partition:

1. Create an ASCII file to use for the SETLINKS utility. The partition and hard disk numbers should match the disk numbers and partitions numbers that you created in "Creating Logical Partitions in an Extended Partition" on page D-9.

You can use a worksheet similar to the one below to assist with the process:

Symbolic Link	Disk No. and Partition Nos.
ACCOUNTING_1	Harddisk1 Partition1
ACCOUNTING_2	Harddisk1 Partition2

2. Run the ASCII input file through the SETLINKS utility:

```
C:\> CD ORACLE_BASE\ORACLE_HOME\BIN
C:\ORACLE_BASE\ORACLE_HOME\BIN> SETLINKS /F:PATH/FILENAME
```

SETLINKS maps the drives to the symbolic label names. Using the sample input file in the above figure, the SETLINKS output looks as follows:

```
Oracle Corporation. Copyright (c) 1999. All rights reserved.
Created Link:
Created Link:6 = Device:\Device\Harddisk1\Partition1
Created Link:ACCOUNTING_2 = Device:\Device\Harddisk1\Partition2
Dos devices updated successfully.
```

3. Ensure that the drives have been mapped with the correct names as shown below:

```
C:\ORACLE_BASE\ORACLE_HOME\BIN> SETLINKS/D
```

SETLINKS shows how the symbolic link names have been mapped:

```
Oracle Corporation. Copyright (c) 1999. All rights reserved.
ACCOUNTING_1  = \Device\Harddisk1\Partition1
ACCOUNTING_2  = \Device\Harddisk1\Partition2
```

Removing or Ignoring Links

You can remove or ignore links by changing the contents of the input file and running it through SETLINKS again.

To remove an existing symbolic link:

■   List a symbolic link without a device path in the input file. For example:

```
ACCOUNTING_1
ACCOUNTING_2 \Device\Harddisk1\Partition2
```

This removes the ACCOUNTING_1 link and creates the other links specified.

To ignore links:

■   Insert a "#" at the beginning of a line to denote a comment, which is ignored by SETLINKS.

Creating a Tablespace in a Raw Partition

To create a tablespace using a data file located in a raw partition:

1.   Start SQL*Plus:

```
C:\> SQLPLUS
```

2.   Connect to the Oracle repository database:

```
Enter user-name: SYSTEM/PASSWORD
```

where *PASSWORD* is MANAGER for the SYSTEM user account by default. If you have changed this password, substitute MANAGER with the correct password.

3.   Create the tablespace. Specify the data file by one of the following:

   ■   If SETLINKS was not used, the drive letter \.*DRIVE_LETTER*:

   ■   If SETLINKS was used, the symbolic link name \\.*SYMBOLIC LINK NAME*

```
SQL> CREATE TABLESPACE TABLESPACE DATAFILE '\\.\DATAFILE' SIZE XM;
```

where:

   ■   *TABLESPACE* is the tablespace name

   ■   '\\.\' is the drive letter or symbolic link name assigned to the raw partition

   ■   *X* is the tablespace size in megabytes (Twenty megabytes is a good starting place.)

For example, to create a tablespace named ACCOUNTING_1 that was assigned a symbolic link name of ACCOUNTING_1, enter the following:

```
SQL> CREATE TABLESPACE ACCOUNTING_1 DATAFILE '\\.\ACCOUNTING_1 SIZE 502M;
```

> **Note:** If you are creating a database with the BUILD_DB.SQL script, modify data files that are stored on raw partitions with a naming convention of \\.*DRIVE_LETTER:* or \\.*SYMBOLIC LINK NAME.* See "Putting the CREATE DATABASE Statement in a Script" on page 6-17.

CRLOGDR Utility

The create logical drives (CRLOGDR) utility allows you to create and delete logical drives and their associated symbolic names on a disk that does not have a primary partition and has only one extended partition.

The CRLOGDR utility is located in the \OPS_PREINSTALL directory on the CD-ROM. To use the utility, copy the CRLOGDR.EXE to a temporary directory.

The utility supports eight parameters, as follows:

/d   Zero indexed drive number (required parameter)

/s   Logical drive size

/n   Logical drive number

/o   Free space offset from beginning of extended partition

/l   Symbolic name

/r   Removes logical drive or symbolic name

/p   Print disk layout

/b   Suppress banner

Examples

The following examples show how to use CRLOGDR.

Example 1

To create a logical drive of 300 MB on disk 3 with no symbolic name:

```
CRLOGDR /d3 /s 300
```

Example 2

To create a logical drive of 100 MB on disk 3 with the symbolic name 'CONTROL_FILE':

```
CRLOGDR /d3 /s 100 /l CONTROL_FILE
```

Example 3

To assign the symbolic name 'DATA_FILE' to the previously created second logical drive (2):

```
CRLOGDR /d3 /n 2 /l DATA_FILE
```

Example 4

To create a logical drive of 300 MB on disk 3 at offset 100 MB with no symbolic name:

> **Note:** The offset /o parameter must be the starting offset of the free space.

```
CRLOGDR /d3 /s 300 /o 100
```

Example 5

To remove second logical drive (2):

> **Note:** This command also removes the symbolic name associated with the drive.

```
CRLOGDR /d3 /r /n 2
```

Example 6

To remove the symbolic name 'CONTROL_FILE' (there is no need to supply disk number):

```
CRLOGDR /r /l CONTROL_FILE
```

Reviewing Disk Layout

You can review the disk layout by using the CRLOGDR utility's print parameter (/p). To print the layout of disk 4, for example:

```
CRLOGDR /d4 /p
Oracle Corporation. Copyright (c) 1998. All rights reserved.
crlogdr - Version 1.0

   NUM        SIZE(MB)       SYMBOLIC NAME  TYPE     OFFSET(MB)
    1             300          DSS_TABLES     6              0
    2             300         OLTP_TABLES     6            300
    3             300          PD_TPCB01      6            600
    4             300          OLTP_INDX      6            900
    5             300           OLTP_RBS      6           1200
    6             300          PD_TPCB04      6           1500
    7             200          OPS_RBS01      6           1800
*Free             5                  --      --           2000

Total Free Space: 5MB
```

The columns have the following significance:

Column	Description
NUM	Logical drive number or free space.
SIZE	Size of logical drive or free space.
SYMBOLIC NAME	Symbolic name associated with the drive, if any. Two hyphens (--) mean no symbolic name has been assigned to the drive.
TYPE	Type of partition, where 6 indicates a large MS-DOS partition and 7 indicates an NTFS partition.
	Note: Raw disk partitions have no file system.
OFFSET	Offset of the partition from the beginning of the extended partition.

E

Net8 Configuration

This appendix describes Net8 configuration for Windows. For an overview of Net8 configuration in general, see the *Net8 Administrator's Guide.*

Specific topics discussed are:

- Unsupported Net8 Features
- Understanding Net8 Registry Parameter and Subkeys
- Listener Requirements
- Understanding Optional Configuration Parameters
- Advanced Network Configuration
- Named Pipes Protocol on Windows 95
- Net8 Port Numbers

> **See:** Net8 integration with Active Directory for Windows 2000 is described in Chapter 4, "Using Oracle8i Directory Server Features with Active Directory".

Unsupported Net8 Features

The following Net8 features are currently unsupported on the Windows NT and Windows 95/98 platforms.

Feature	Description
TRCROUTE	Client application not supported.
SPAWN	SPAWN command in the listener control utility is not supported.
Pre-SPAWNED dedicated server processes	Not supported by the listener. Therefore, do not include the following parameters in the SID_DESC's of the LISTENER.ORA file: PRESPAWN_MAX, PROTOCOL, POOL_SIZE, and TIMEOUT.

Understanding Net8 Registry Parameter and Subkeys

Net8 contains the registry entries for Net8 parameters and Net8 service subkeys. To successfully add or modify Net8 configuration parameters, you must understand where they are located and the rules that apply to them.

Net8 Parameters

The location of the Oracle Net8 registry parameters is:

HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME*ID*

where *ID* is incremented for each additional Oracle home directory on your computer (for example, HOME0 is for a first directory, HOME1 is for a second directory, and so forth).

Net8 Service Subkeys

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\SERVICES` contains subkeys that correspond to services. Depending on what is installed, your Net8 services will consist of all or a subset of the following:

- Oracle*HOME_NAME*ClientCache

- Oracle*HOME_NAME*CMAdmin

- Oracle*HOME_NAME*CMan

- Oracle*HOME_NAME*TNSListener

Each service subkey contains the following parameters:

Parameter	Description
DisplayName	Specifies the service name.
ImagePath	Specifies the fully qualified path name of the executable invoked by the service and any command line arguments passed to the executable at runtime.
ObjectName	Specifies the logon user account and computer to which the service should log on.

Listener Requirements

A release 8.1.6 listener is required for the Oracle8*i* release 8.1.6 database. Previous versions of the listener are not supported with the Oracle8*i* release 8.1.6 database. However, the release 8.1.6 listener is supported with previous versions of the database.

Once Oracle8*i* is installed, Oracle Corporation recommends that you use the release 8.1.6 listener for all of your Oracle release 8.1 databases and previous releases (such as an Oracle8 8.0 database). Even if you install Oracle8*i* in multiple Oracle homes on the same computer, you should only use one listener for all your databases on the same computer.

Additionally, the new 8.1.6 listener is set to start automatically at system reboot. If you intend to use only the 8.1.6 listener for all of your databases, ensure that only the Windows NT service for the 8.1.6 listener, as listed in the Windows NT services Control Panel, is set to start automatically.

If you must use multiple listeners (such as a Net8 8.0 listener and Net8 8.1 listener) on the same computer, see the *Net8 Administrator's Guide* and Chapter 5 for more information.

If the 8.1.6 listener service is not created during Oracle8*i* installation due to conflicts in listening endpoints, you can start your 8.1.6 listener by using the LSNRCTL control utility:

To start the listener using LSNRCTL:

```
C:\> LSNRCTL START LISTENER_NAME
```

where *LISTENER_NAME* is the listener name for typical install or a name given during custom install.

This command also creates the NT service for the listener if the service does not already exist.

To exit the utility:

```
LSNRCTL> EXIT
```

In past releases, information about instances was configured manually in the LISTENER.ORA file. Instance registration is now automatic. Instances register themselves with the listener when they are started.

Service Registration Not Supported over IPC

The Windows NT platform does not support the IPC adapter for multithreaded server or service registration. An Oracle8*i* database automatically registers certain information with the listener. This feature, called service registration, is not supported over IPC on Windows NT. This is because the IPC adapter, Named Pipes, does not support asynchronous notification. For the same reason, dispatchers do not work with the IPC adaptor on Windows NT.

Understanding Optional Configuration Parameters

You can use the following parameters on Windows NT and Windows 95/98:

- LOCAL

- TNS_ADMIN

- USE_SHARED_SOCKET

Net8 first checks for the parameters as environment variables, and uses the values defined. If environment variables are not defined, it searches for these parameters in the registry.

> **Additional Information:** See "Appendix C, Oracle8i Configuration Parameters and the Registry " for instructions on editing Windows registry keys.

LOCAL

You can add the LOCAL parameter to make a connection without specifying a connect string service name. The value for LOCAL is the service name in the `TNSNAMES.ORA` file located in the *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN directory.

For example, if the LOCAL parameter is specified as finance, you connect to a database from SQL*Plus with the following command:

```
SQL> CONNECT SCOTT/TIGER
```

Net8 checks if LOCAL is defined as an environment variable or as a parameter in the registry, and uses finance as the service name. If it exists, Net8 connects.

TNS_ADMIN

You can add the TNS_ADMIN parameter to change the directory name for configuration files from the default location. For example, if you set TNS_ADMIN to *ORACLE_BASE\ORACLE_HOME*\TEST\ADMIN, the configuration files are used from *ORACLE_BASE\ORACLE_HOME*\TEST\ADMIN.

USE_SHARED_SOCKET

You can set the USE_SHARED_SOCKET parameter to TRUE to enable the use of shared sockets. If this parameter is set to TRUE, the network listener passes the socket descriptor for client connections to the database thread. As a result, the client does not need to establish a new connection to the database thread and database connection time improves. Also, all database connections share the port number used by the network listener, which can be useful if you are setting up third-party proxy servers.

On Windows NT 4.0 Service Pack 3 or earlier, enabling this option precluded bringing the network listener up or down in a case where a database connection spawned by the network listener is active. This is not an issue on Windows NT 4.0 Service Pack 4 or later. Oracle recommends that you upgrade to Windows NT 4.0 Service Pack 4 if you intend to set this parameter.

This parameter only works in dedicated server mode in a TCP/IP environment. If this parameter is set, you cannot use the 8.1.6 listener to spawn Oracle 7.*x* databases. To spawn an Oracle 8.0.*x* database from an 8.1.6 listener with the shared socket enabled, you must also set the variable *USE_SHARED_SOCKET* for the 8.0.*x* Oracle home.

Advanced Network Configuration

The following sections describe advanced configuration procedures specifically for Net8 on the Windows NT and Windows 95/98 platforms.

Configuring Authentication Methods

Net8 provides two authentication methods for Windows NT and for Windows 95/98, NDS Authentication and Windows Native Authentication.

NDS Authentication

The automatically installed Novell Directory Service (NDS) authentication method allows client applications and users to access a NetWare server running Oracle through NDS. Following NDS authentication, a user logged into an NDS directory tree can use an Oracle database on a NetWare server in the same tree. This permits the user from having to enter an additional user name and password.

> **Note:** To connect from a client using the NDS authentication method, the server must be running the NetWare operating system.

If you also use NDS External Naming, you can view the entire network under a single NDS directory tree.

Configuring a NetWare Server

> **Note:** SQLNET.AUTHENTICATION_SERVICES enables one or more authentication services. If authentication has been installed, it is recommended that this parameter be set to either NONE or to one of the authentication methods.

To configure the server:

1. Install and configure Net8 for NetWare on the server.

2. Add the following entry to the SQLNET.ORA file:

```
SQLNET.AUTHENTICATION_SERVICES = (NDS)
```

Configuring a Client

To configure a client:

1. Install and configure the Net8 on a client.

2. Add the following entry to the SQLNET.ORA file:

```
SQLNET.AUTHENTICATION_SERVICES = (NDS)
```

Connecting from a Client

To connect from a client with the NDS authentication method:

1. Log into the NDS tree.

2. Enter the following command to access an Oracle for NetWare database:

```
C:\> SQLPLUS
SQL> CONNECT /@SERVICE_NAME
```

> **See:** Chapter 8, "Authenticating Database Users with Windows" for instructions on configuring and using Windows Native Authentication methods.

Using the NDS Naming Method

NDS naming refers to the resolution of a service name by using a supported third-party naming service. The NDS Native Naming Adapters resolve service names stored in a native naming service.

> **Note:** To connect from a client using the NDS Native Naming Adapter, the server must be running the NetWare operating system.

The NDS Native Naming Adapter for Windows NT and Windows 95/98 clients uses the NDS naming environment to store service names and addresses of Oracle8*i* NetWare Servers. This lets an NDS user view the entire network under a single NDS directory tree. You can use native name services in addition to, or instead of, Oracle Names or the TNSNAMES.ORA file.

If the NDS Authentication Adapter is used as well, a single logon can access a multi-server and multi-database network.

NetWare Server Configuration

To configure the NetWare Server:

1. Install and configure Net8 for NetWare on your server.

2. Log into the NDS tree.

3. Add NOVELL to the NAMES.DIRECTORY_PATH parameter in the SQLNET.ORA file:

```
NAMES.DIRECTORY_PATH = (NOVELL, TNSNAMES, ONAMES)
```

Client Configuration

To configure the client:

1. Install and configure the NDS Native Naming Adapter and Net8 on your client.

2. When you configure the NDS Native Naming Adapter, add NOVELL to the NAMES.DIRECTORY_PATH parameter in the SQLNET.ORA file:

   ```
   NAMES.DIRECTORY_PATH = (NOVELL, TNSNAMES, ONAMES)
   ```

Client Connection

To connect from a client with the NDS Native Naming Adapter:

Enter the following command to access an Oracle8 for NetWare database:

```
C:/> SQLPLUS
SQL> CONNECT USERNAME/PASSWORD@DATABASE_OBJECT_NAME
```

where *DATABASE_OBJECT_NAME* identifies Oracle8*i* in NDS.

> **See:**
> - Novell NetWare documentation for further information about NDS
> - Oracle8*i* NetWare documentation

Configuring Security for Named Pipes Protocol

If you are using Named Pipes protocol with Oracle Names, the network listener may not be able to connect to the Oracle Names server.

Oracle Names creates a "named pipe" at startup time. The network listener tries to open the Named Pipe at startup. If it cannot, the network listener uses the default system account "Local System."

The network listener service may be unable to open the Named Pipe created by Oracle Names unless the Oracle*HOME_NAME*TNSListener service has a valid user ID and password associated.

To set up the network listener permissions:

1. From the **Control Panel** window, double-click **Services.**

    The **Services** window appears.

2. Select the Oracle*HOME_NAME*TNSListener service and double-click.

    The *Services* dialog box appears.

3. Click the This Account option button. Then, click the **"..."** option button next to it.

    The *Add User* dialog box appears.

4. Select your logon ID (user ID) from the Names list and click Add**.**

    The user ID appears in the Add Name text box.

5. Click **OK.**

    The *Services* dialog box appears with the user ID displayed in the This Account text box.

6. Type your password in the Password text box.

7. Retype the same logon password in the Confirm Password text box.

8. Click OK**.**

Named Pipes Protocol on Windows 95

If you use the Named Pipes protocol on a Windows 95 system to connect to Oracle8*i* for Windows NT, client applications may run very slowly due to a known problem in Microsoft's implementation of Windows 95 NWLinkDirect-Hosting.

To work around this problem, you may do any of the following:

- Use other protocols (for exampl, TCP/IP, SPX) for connecting from an Oracle client

- Remove the protocol NWLink from the Windows 95 system if you do not need to access NetWare Servers

- Disable Direct-Hosting feature on Windows 95

Refer to Microsoft documentation for detailed information.

Net8 Port Numbers

The following table describes the port numbers used by Net8.

Product	Default Port Number	How do I change the port number?
Listener	1521	Modify the LISTENER.ORA and TNSNAMES.ORA files, using Net8 Assistant. These files are located in the *ORACLE_BASE**ORACLE_HOME*\\NETWORK\\ADMIN directory, or in the directory specified by the TNS_ADMIN environment variable or registry value.
Oracle Names	1575	Modify the NAMES.ORA file, using Net8 Assistant. NAMES.ORA is located in the *ORACLE_BASE**ORACLE_HOME*\\NETWORK\\ADMIN directory.
Oracle Connection Manager, using TCP/IP	1630 1830	Modify the CMAN.ORA configuration file, using Net8 Assistant. CMAN.ORA is located in the *ORACLE_BASE**ORACLE_HOME*\\NETWORK\\ADMIN directory, or in the directory specified by the TNS_ADMIN environment variable or registry value.

F

SNMP Support

This appendix describes how to use Oracle SNMP Agent in Net8 for Windows NT.

Specific topics discussed are:

- What is the Purpose of SNMP?
- Oracle SNMP Support
- Oracle SNMP Agent for Oracle Services
- Configuring Oracle SNMP Agent
- Controlling the Master Agent and the Encapsulator
- Understanding the MASTER.CFG File
- Understanding the ENCAPS.CFG File

Additional Information: For a complete description of SNMP concepts and terms, detailed listings of management information contents, and suggestions on how to use SNMP effectively to develop management applications, see the *Oracle SNMP Support Reference Guide.*

What is the Purpose of SNMP?

Simple Network Management Protocol (SNMP) is a standard internet protocol enabling certain nodes in a network (the management stations or managing nodes) to query other network components or applications for information about their status and activities. Such a query is known as an SNMP poll. These items that can be polled are called managed elements.

The software used by a management station is called a management framework or management station. The management station uses the SNMP protocol to request information from subagents on the nodes being managed, and those agents send back the appropriate responses. The agents can also transmit messages called traps independently from the framework to well–known addresses in response to specific events. This enables quick and sometimes automatic reactions to the specific conditions indicated by the traps.

All requests sent to a given network node are handled by the same master agent. This agent redirects the requests to the appropriate managed elements on the node, in some cases using subagents. The information that SNMP can obtain is described in a structure called a Management Information Base (MIB), which is located on the node of the managed element.

Oracle SNMP Support

Oracle SNMP support enables Oracle products such as Oracle8*i* for Windows NT to be located, identified, and monitored by a management station running at one or more centrally located nodes.

SNMP support allows a database to be remotely monitored by any SNMP-capable management software in a TCP/IP network. This feature enables key Oracle products running anywhere on an enterprise's network to be located, identified, and monitored by a management station running at a centrally located node. This is done in much the same way and using many of the same tools as have traditionally been used to monitor the activity of the network itself. Oracle SNMP thereby integrates the tasks of database administrators and network administrators, enabling both to use some of the same tools and to better integrate their tasks.

> **Note:** For this release, Oracle Networking products (Oracle Listener and Oracle Names) do not support SNMP. Therefore, the corresponding MIBs are not supported, but Oracle*8i* for Windows NT release 8.1.6 supports SNMP and the database MIB is supported.

Oracle SNMP Agent for Oracle Services

Oracle SNMP Agent allows you to configure an:

- OracleSNMPPeerMasterAgent
- OracleSNMPPeer Encapsulator

OracleSNMPPeerMasterAgent The OracleSNMPPeerMasterAgent is the process on a managed node that accepts queries from the management framework and communicates with the subagents to answer the query. It also can send SNMP traps independently in response to specific conditions. Only one master agent can exist on each managed node. Any node that does not have an agent will not be able to respond to SNMP requests.

OracleSNMPPeerEncapsulator If you are currently using an incompatible agent, such as Microsoft SNMP, you must install the Encapsulator to encapsulate the Microsoft master agent such that all SNMP requests from a Network Management Station (NMS) are sent to OracleSNMPPeerMasterAgent, which then forwards the relevant requests to the encapsulated master agent through the encapsulator. If the encapsulator is not configured, Microsoft SNMP will not work.

> **Note:** MIB files for Oracle products are installed in the *ORACLE_BASE\ORACLE_HOME*\NETWORK\DOCS directory. The Oracle SNMP agent can be installed in only one Oracle home on a system.

Configuring Oracle SNMP Agent

This section briefly describes the major tasks that the network administrator must perform to enable the Oracle SNMP Agent.

Oracle SNMP Agent is installed with Oracle Universal Installer.

To configure Oracle SNMP support on a managed node:

1. Specify the port where the master agent is listening. (For information on port numbers, see "Net8 Port Numbers" on page E-11.)

   The port is specified in the TRANSPORT section of the MASTER.CFG file located at *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN.

   For example, add the following section to the file:

   ```
   TRANSPORT        ordinary        SNMP
                    OVER UDP SOCKET
                    AT PORT 161
   ```

   > **Note:** Oracle Corporation recommends the use of port 161 for OracleSNMPPeerMasterAgent since it is the default port for SNMP communication. However, you may specify a different port as long as the management application on your NMS can be configured to send SNMP requests to the master agent listening on this port.

2. Specify the authentication in the COMMUNITY section of the MASTER.CFG file:

   ```
   COMMUNITY    public
                ALLOW ALL OPERATIONS
                USE NO ENCRYPTION
   ```

3. Continue to Step 4 if the Encapsulator is to be used.

4. Specify an unused port where the encapsulated agent, Microsoft SNMP Service, should be listening.

   The port is specified in the SERVICES file located at *NT_HOME*\SYSTEM32\DRIVERS\ETC.

   For example, make sure you have the following line in the file:

   ```
   snmp              1161/udp    snmp
   ```

   > **Note:** If there has already been an entry for SNMP in the file, change the port from 161 (default number) to something else that is available (1161 in this example).

5. Edit the Encapsulator configuration file, ENCAPS.CFG, located at *ORACLE_BASE**ORACLE_HOME*\NETWORK\ADMIN to specify which non-PEER master agents are to be encapsulated.

   You must at least add an AGENT entry, including MIB subtrees manageable by NMS, for the encapsulated master agent.

   For example, you should have a section like the following in the file:

   ```
   AGENT AT PORT 1161 WITH COMMUNITY public
   SUBTREES

   1.3.6.1.2.1.1,
   1.3.6.1.2.1.2,
   1.3.6.1.2.1.3,
   1.3.6.1.2.1.4,
   1.3.6.1.2.1.5,
   1.3.6.1.2.1.6,
   1.3.6.1.2.1.7,
   1.3.6.1.2.1.8,
   1.3.6.1.2.1.77

   FORWARD ALL TRAPS;
   ```

   > **Note:** The port (1161 in this example) must match the one you specified in Step 4.

Controlling the Master Agent and the Encapsulator

Starting the Master Agent

You may start the master agent from the MS-DOS command prompt or from the Windows NT Control Panel's *Services* dialog box. Both ways are described below.

To start the master agent from the MS-DOS command prompt:

Invoke the master agent as follows:

```
C:\> CD ORACLE_BASE\ORACLE_HOME\BIN
C:\> AGENT CONFIGURATION_FILE TEMPORARY_FILE
```

where *CONFIGURATION_FILE* is the name of the master configuration file and *TEMPORARY_FILE* is a temporary file where storage information is to be stored. The master configuration file name is defaulted to MASTER.CFG.

> **Note:**   *CONFIGURATION_FILE* and *TEMPORARY_FILE* are mandatory.

To start the master agent:

1. From the *Control Panel* window, double-click Services**.**

    The *Services* dialog box appears. Look for the OracleSNMPPeerMasterAgent service. A blank in Status column indicates that the service has stopped.

    If the master is running, skip to Step 3. If the master is not running, continue to Step 2.

2. Select the OracleSNMPPeerMasterAgent service, and click Start**.**

    The master agent starts.

3. In the *Services* dialog box, choose the Close button.

Starting the Encapsulator

You may start the encapsulator from the MS-DOS command prompt or from the Windows NT Control Panel's *Services* dialog box. Both ways are described below.

> **Note:**   If the Encapsulator is to be used, the master agent must be started first. See "Starting the Master Agent" in this appendix.

To start the encapsulator from the MS-DOS command prompt:

```
C:\> CD ORACLE_BASE/ORACLE_HOME\BIN
C:\> ENCAPS [options]
```
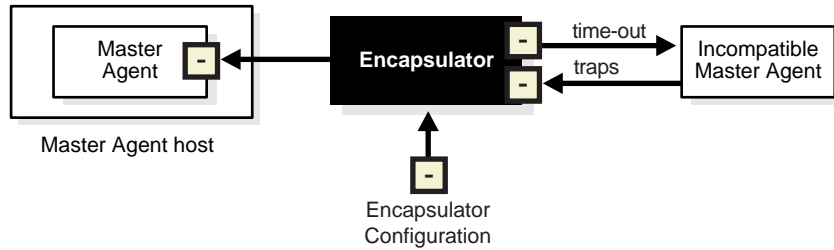

There are six command line options as shown below. The options are:

-T *PORT*	Specifies the incoming trap port where the encapsulator listens to receive SNMP traps sent by encapsulated master agents.
	Default: no traps forwarded
-S *PORT*	Specifies the port from which the encapsulator sends SNMP requests to encapsulated agents, and to which the encapsulated agents send their responses.
	Default: assigned by the host
-H *HOST*	Specifies the master agent's host address, that is, the host where the master agent resides. This may or may not be the host of the encapsulated agent, specified in the ENCAPS.CFG file.
	Default: localhost
-P *PORT*	Specifies the master agent's listening SMUX port, that is, the port to which the encapsulator sends its SMUX traffic.
	Default: 199
-W *WAIT*	Specifies the SNMP response time-out (in seconds) from the encapsulated master agent.
	Default: 2 s
-C *FILE*	Specifies the encapsulator's configuration file. This file is defaulted to ENCAPS.CFG.

To start the encapsulator from the Control Panel Window:

1. From the *Control Panel* window, double-click Services**.**

   The *Services* dialog box appears. Look for the OracleSNMPPeerEncapsulator service. A blank in the Status column indicates that the service is stopped.

   If the master is running, skip to Step 3. If the master is not running, continue to Step 2.

2. Select the OracleSNMPPeerEncapsulator service, and click Start**.**

   The encapsulator is started.

3. In the *Services* dialog box, choose the Close button.

Understanding the MASTER.CFG File

The MASTER.CFG file controls several aspects of the OracleSNMPPeerMaster Agent, including:

- Choice of transport protocols
- Community-based access control
- Community-based naming

> **Note:** The MASTER.CFG file is automatically installed on the server at *ORACLE_BASE\ORACLE_HOME*\NETWORK\ADMIN.**:**

A sample MASTER.CFG file is shown in the figure below:

```
########################################################################
# Default Agent Configuration File
#
#       This file allows MANAGERS to be specified.  This is used to
#       specify which managers will be receiving which traps.
#
#       Also, COMMUNITYs can be specified. This allows that agent to
#       be configured such that it will only except requests from
#       certain managers and with certain community strings.
#
# Syntax
# TRANSPORT <name>   SNMP
#                  [OVER UDP SOCKET]
#                  [AT <addr>]
# COMMUNITY <communityName>
#              ALLOW <op> [,<op>]*  [OPERATIONS]
#                  [AS ENTITY <entityName>]
#                  [MEMBERS <addrs> [,<addrs>] ]
TRANSPORT    ordinary       SNMP
   OVER UDP SOCKET
   AT PORT 161
COMMUNITY    public
   ALLOW ALL OPERATIONS
   USE NO ENCRYPTION
```

Below is a description of the TRANSPORT and COMMUNITY parameters.

TRANSPORT	Defines an interface over which the master agent listens for SMUX connections from subagents or SNMP requests.
COMMUNITY	Provides a form of authentication for access to information.
	Note: This parameter is not the same as the COMMUNITY parameter in the TNSNAMES.ORA file.

Understanding the ENCAPS.CFG File

The ENCAPS.CFG configuration file specifies which incompatible agents are to be encapsulated and what to make visible to the Network Management Station (NMS).

A sample ENCAPS.CFG configuration file is shown below:

```
############################################################################
#
#        THIS IS AN EXAMPLE OF AN ENCAPSULATOR CONFIGURATION FILE.
#
#        USING THIS FILE, ENCAPSULATOR WILL ENCAPSULATE A NON-PEER
#        MIB2 AGENT ON THE SAME PROCESSOR, THAT'S LISTENING FOR SNMP
#        REQUESTS AT PORT 1161.
#
#        IT ALLOWS THE NETWORK MANAGEMENT STATION TO VIEW THE 8 SUBTREES
#        MANAGED BY THAT AGENT, AND TO RECEIVE ALL THE TRAPS EMITTED BY
#        THAT AGENT.  ALL THE REQUESTS AND TRAPS PASS THROUGH THE PEER
#        AGENT AND ENCAPSULATOR.  AT THE SAME TIME THAT THE PEER AGENT
#        IS HANDLING ENCAPSULATOR, IT ALSO HANDLES ALL SUB-AGENTS THAT
#        WERE DEVELOPED WITH THE PEER TOOLKIT TO MANAGE OTHER MIBS.
# Syntax:
# [
# AGENT  [ON HOST <ip>]  [AT PORT <port>]  [WITH COMMUNITY <community>]
# SUBTREES <treelist>
# [FORWARD <traplist> TRAPS]
# ;
# ]+
#
AGENT AT PORT 1161 WITH COMMUNITY public
SUBTREES


1.3.6.1.2.1.1,
1.3.6.1.2.1.2,
1.3.6.1.2.1.3,
1.3.6.1.2.1.4,
1.3.6.1.2.1.5,
1.3.6.1.2.1.6,
1.3.6.1.2.1.7,
1.3.6.1.2.1.8,
1.3.6.1.4.1.77


FORWARD ALL TRAPS;
```

Below is a description of the parameters.

AGENT	Defines which incompatible master agents are encapsulated. Each AGENT entry defines the port on which the encapsulated master agent is listening for incoming SNMP requests and which community string Oracle Peer SNMP should use on SNMP requests sent to that agent.
SUBTREES	Identifies which of the subtrees managed by the encapsulated agent should be made visible to the NMS by listing the object identifiers.
FORWARD TRAPS	Defines which traps sent by encapsulated agents on a host will be forwarded by Oracle Peer SNMP to Encapsulator master agent.

Installing Oracle SNMP Agent with Other Network Management Systems

Users of other network management systems, such as HP OpenView, may encounter interference with the encapsulator if they are installed on the same machine after installation of Oracle SNMP Agent. This results from an HP OpenView change to the SNMP port in the SERVICES file. The port specified for SNMP in the SERVICES file must match that specified in the ENCAPS.CFG file.

To resolve the problem, after installing HP Open View, change the SNMP port in either the SERVICES file or the ENCAPS.CFG, so that the two files agree.

G

Error Messages

This appendix lists the error messages, causes, and corrective actions that are specific to the operation of Oracle8*i* for Windows NT. This appendix also includes database connection issues.

Specific topics discussed are:

- Logging Error Messages
- Codes 04000-04999: Windows NT-Specific Oracle Messages
- Database Connection Issues

> **Note:** The ORA.HLP file, which was shipped in previous releases, is no longer available. See this Appendix and *Oracle8i Error Messages* for information on error messages.

Logging Error Messages

Keep a log of error messages you receive by redirecting the messages to a file. You can record the contents of normal utility messages by using the LOGFILE parameter discussed in *Oracle8i Utilities*. You can separately record the error message portion by using standard Windows NT file redirection. For example, use the following syntax to redirect the output from the Export Utility:

```
C:\> EXP USERNAME/PASSWORD PARFILE=FILENAME >FILE1.LOG 2>FILE2.ERR
```

In this command line, FILE1.LOG receives the standard output from Export, while FILE2.ERR receives the standard error messages.

Codes 04000-04999: Windows NT-Specific Oracle Messages

The error messages in this section are Oracle operating system-dependent (OSD) messages issued in response to an error condition in Windows NT. Each message in this section triggers an Oracle8i database error message.

- File I/O Errors: OSD-04000 to OSD-04099

- Memory Errors: OSD-04100 to OSD-04199

- Process Errors: OSD-04200 to OSD-04299

- Loader Errors: OSD-04300 to OSD-04399

- Semaphore Errors: OSD-04400 to OSD-04499

- Miscellaneous Errors: OSD-04500 to OSD-04599

File I/O Errors:	OSD-04000 to OSD-04099
4000	logical block size mismatch
4001	invalid logical block size
4002	unable to open file
4003	unable to read file header block
4004	invalid file header
4005	SetFilePointer() failure, unable to read from file
4006	ReadFile() failure, unable to read from file
4007	truncated read
4008	WriteFile() failure, unable to write to file

File I/O Errors:	OSD-04000 to OSD-04099
4009	truncated write
4010	<create> option specified, file already exists
4011	GetFileInformationByHandle() failure, unable to obtain file info
4012	file size mismatch
4013	unable to read line from file
4014	unable to close file
4015	An asynchronous I/O request returned an error
4016	Error queuing an asynchronous I/O request
4017	Unable to open the specified RAW device
4018	Unable to access the specified directory or device
4019	Unable to set file pointer
4020	Unable to set eof file marker
4021	Unable to read file
4022	Unable to write file
4023	SleepEx() failure, unable to Sleep
4024	Unable to delete file
4025	Invalid question asked
4026	Invalid parameter passed

Memory Errors:	OSD-04100 to OSD-04199
4100	malloc() failure, unable to allocate memory
4101	invalid SGA: SGA not initialized
4102	Unable to open/create file for shared memory objec
4103	unable to attach to SGA: SGA does not exist
4104	Unable to map shared memory (SGA) into the address space
4105	Shared memory (SGA) mapped to wrong address
4106	Unable to allocate memory with VirtualAlloc
4107	Unable to deallocate memory with VirtualFree
4108	Unable to protect memory with VirtualProtect

Process Errors:	OSD-04200 to OSD-04299
4200	unable to begin another thread
4201	no pid structure supplied to spdcr()
4202	DosSetPriority() failure, unable to set process priority
4203	DosKillProcess() failure, unable to kill process
4204	invalid pid
4205	CreateProcess() failure, unable to spawn process
4207	invalid priority specified in CONFIG parameter ORACLE_PRIORITY
4208	OpenProcess() failure, unable to open process handle
4209	Incorrect or unknown backgound image name given to spdcr()
4210	Timeout waiting for thread semaphore
4211	Thread information not found
4212	Maximum number of ORACLE threads reached
4213	ORACLE thread unable to DuplicateHandle()
4214	ORACLE thread unable to CreateEvent()
4215	Bad function code supplied to ssthreadop
4216	Unable to find file handle for that thread
4217	Unable to retrieve system user name for current user
4218	Can not post thread
4219	Bad thread list semaphore
4221	Target thread is currently busy
4222	Unable to get the threads context
4223	Unable to set the threads context
4224	Unable to suspend the target thread
4225	Unable to resume the target thread

Loader Errors:	OSD-04300 to OSD-04399
4300	unable to read complete record from data file
4301	record size too large
4302	invalid record type and/or load options

Semaphore Errors:	OSD-03400 to OSD-03499
4400	unable to acquire internal semaphore for process
4401	WaitForSingleObject() failure, unable to obtain semaphore

Miscellaneous Errors:	OSD-04500 to OSD-04599
4500	illegal option specified
4501	internal buffer overflow
4502	translations nested too deep
4503	text contains no translatable elements
4505	stdin not responding
4506	unable to spawn process via system()
4507	password for 'internal' is incorrect
4508	no password given
4509	no password found
4510	operating system roles are not supported
4511	unable to get date and time from the operating system
4512	unable to translate the 'USERNAME' config.ora variable on server
4513	'remote_os_authent' init.ora variable not set to true
4514	The NT Group name is too long for internal buffer
4515	This command is not implemented at this time

File I/O Errors: OSD-04000 to OSD-04099

OSD-04000

Logical block size mismatch

Cause: The database block size specified in the initialization parameter file does not match the block size of the actual database files.

Action: Use matching logical block sizes.

OSD-04001

Invalid logical block size

Cause: The logical block size is not a multiple of 512 bytes, or it is too large.

Action: Change the value of DB_BLOCK_SIZE in the initialization parameter file.

OSD-04002

Unable to open file

Cause: The specified path or file name is invalid, or the destination device is full. This error can also be caused by insufficient Windows NT file handles.

Action: Make sure the path and file exist, and the device has free space. If this fails, increase the number of Windows NT file handles.

OSD-04003

Unable to read file header block

Cause: The media has been damaged.

Action: Recover the file if necessary, and verify that Windows NT is functioning correctly.

OSD-04004

Invalid file header

Cause: The file is corrupted.

Action: Recover the file.

OSD-04005

SetFilePointer() failure, unable to read from file

Cause: There was an unexpected return from the Windows NT system service, SetFilePointer().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04006

ReadFile() failure, unable to read from file

Cause: There was an unexpected return from the Windows NT system service, ReadFile().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04007

Truncated read

Cause: The system encountered an unexpected end-of-file, which is due to damaged media.

Action: Verify that the file is not damaged.

OSD-04008

WriteFile() failure, unable to write to file

Cause: There was an unexpected return from the Windows NT system service, WriteFile().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04009

Truncated write

Cause: The destination device is full or the media is damaged.

Action: Verify that the device has free space and the file is not damaged.

OSD-04010

<create> option specified, file already exists

Cause: The file you attempted to create already exists.

Action: Delete the existing file or use the REUSE option in the SQL statement.

OSD-04011

GetFileInformationByHandle() failure, unable to obtain file info

Cause: There was an unexpected return from the Windows NT system service, GetFileInformationByHandle().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04012

File size mismatch

Cause: The file to be re-used is either too large or too small.

Action: Specify the correct file size or delete the existing file.

OSD-04013

Unable to read line from file

Cause: This error is caused by an operating system error or by damaged media.

Action: Check the operating system error code (if available) and consult the Windows NT documentation. If no operating system error code is presented, verify that the media is not damaged.

OSD-04014

Unable to close file

Cause: The media has been damaged.

Action: Recover the file, if necessary, and verify that Windows NT is functioning correctly.

OSD-04015

Asynchronous I/O request returned an error

Cause: There was an unexpected return from the Windows NT system service.

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04016

Error queuing an asynchronous I/O request

Cause: There was an unexpected return from the Windows NT system service.

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04017

Unable to open the specified RAW device

Cause: An invalid path or file name was specified or the device is full.

Action: Make sure the file exists and/or device is not full; verify that the operating system is functioning correctly.

OSD-04018

Unable to access the specified directory or device

Cause: An invalid path name was specified.

Action: Make sure the directory or device exists and is accessible.

OSD-04019

Unable to set file pointer

Cause: This error is caused by an operating system error or by damaged media.

Action: Check the operating system error code (if available) and consult the Windows NT documentation. If no operating system error code is presented, verify that the media is not damaged.

OSD-04020

Unable to set eof file marker

Cause: This error is caused by an operating system error or by damaged media.

Action: Check the operating system error code (if available) and consult the Windows NT documentation. If no operating system error code is presented, verify that the media is not damaged.

OSD-04021

Unable to read file

Cause: This error is caused by an operating system error or by damaged media.

Action: Check the operating system error code (if available) and consult the Windows NT documentation. If no operating system error code is presented, verify that the media is not damaged.

OSD-04022

Unable to write file

Cause: This error is caused by an operating system error or by damaged media.

Action: Check the operating system error code (if available) and consult the Windows NT documentation. If no operating system error code is presented, verify that the media is not damaged.

OSD-04023

SleepEx() failure, unable to Sleep

Cause: There was an unexpected return from the Windows NT system service.

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04024

Unable to delete file

Cause: This error is caused by an operating system error or by damaged media.

Action: Check the operating system error code (if available) and consult the Windows NT documentation. If no operating system error code is presented, verify that the media is not damaged.

OSD-04025

Invalid question asked

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04026

Invalid parameter passed

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

Memory Errors: OSD-04100 to OSD-04199

OSD-04100

Malloc() failure, unable to allocate memory

Cause: The program is out of memory.

Action: Shut down all unnecessary processes or install more memory in the computer.

OSD-04101

Invalid SGA: SGA not initialized

Cause: The System Global Area (SGA) has been allocated but not initialized.

Action: Wait until the STARTUP has completed before attempting to connect.

OSD-04102

Unable to open/create file for shared memory object

Cause: There was an unexpected return from the Windows NT system service, CreateFile().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04103

Unable to attach to SGA: SGA does not exist

Cause: The SGA does not exist.

Action: Start up an Oracle instance.

OSD-04104

Unable to map shared memory (SGA) into the address space

Cause: There was an unexpected return from the Windows NT system service, MapViewOfFileEx().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04105

Shared memory (SGA) mapped to wrong address

Cause: There was an unexpected return from the Windows NT system service, MapViewOfFileEx().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04106

Unable to allocate memory with VirtualAlloc

Cause: The program is out of memory.

Action: Shut down all unnecessary processes or install more memory in the computer.

OSD-04107

Unable to deallocate memory with VirtualFree

Cause: There was an unexpected return from the Windows NT system service, VirtualFree().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04108

Unable to protect memory with VirtualProtect

Cause: There was an unexpected return from the Windows NT system service, VirtualProtect().

Action: Check the operating system error code and consult the Windows NT documentation.

Process Errors: OSD-04200 to OSD-04299

OSD-04200

Unable to begin another thread

Cause: The program has run out of system resources.

Action: Shut down all unnecessary processes; install more memory in the computer.

OSD-04201

No pid structure supplied to spdcr()

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04202

DosSetPriority() failure, unable to set process priority

Cause: There was an unexpected return from the Windows NT system service, DosSetPriority().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04203

DosKillProcess() failure, unable to kill process

Cause: There was an unexpected return from the Windows NT system service, DosKillProcess().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04204

Invalid pid

Cause: Process ID not recognized by system, process previously terminated.

Action: Verify that process ID is correct and that process is active.

OSD-04205

CreateProcess() failure, unable to spawn process

Cause: There was an unexpected return from the Windows NT system service, CreateProcess().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04207

Invalid priority specified in CONFIG parameter ORACLE_PRIORITY

Cause: The priority specified is invalid or out of range.

Action: Specify a valid setting for ORACLE_PRIORITY.

OSD-04208

OpenProcess() failure, unable to open process handle

Cause: There was an unexpected return from the Windows NT system service, OpenProcess().

Action: Check the operating system error code and consult the Windows NT documentation.

OSD-04209

Incorrect or unknown background image name given to spdcr()

Cause: There was an unexpected background name given to spdcr().

Action: Contact Oracle Support Services.

OSD-04210

Timeout waiting for thread semaphore

Cause: An Oracle8 database thread died holding the semaphore.

Action: Restart Oracle8 database instance.

OSD-04211

Thread information not found

Cause: An Oracle8 database thread died without deleting its information.

Action: Restart Oracle8 database instance.

OSD-04212

Maximum number of Oracle threads reached

Cause: The maximum number of Oracle8 database threads for the instance is reached.

Action: Wait until some connections exit before trying again.

OSD-04213

Oracle thread unable to DuplicateHandle()

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04214

Oracle thread unable to CreateEvent()

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04215

Bad function code supplied to ssthreadop

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04216

Unable to find file handle for that thread

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04217

Unable to retrieve system username for current user

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04218

Cannot post thread

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04219

Bad thread list semaphore

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04221

Target thread is currently busy

Cause: The target thread is processing an oradebug command.

Action: Wait and re-issue command.

OSD-04222

Unable to get the threads context

Cause: Check OS error code.

Action: Remedy OS error.

OSD-04223

Unable to set the threads context

Cause: Check OS error code.

Action: Remedy OS error.

OSD-04224

Unable to suspend the target thread

Cause: Check OS error code.

Action: Remedy OS error.

OSD-04225

Unable to resume the target thread

Cause: Check OS error code.

Action: Remedy OS error.

Loader Errors: OSD-04300 to OSD-04399

OSD-04300

Unable to read complete record from data file

Cause: The data file ended in the middle of a record. This error occurs when loading files with a fixed record length.

Action: Verify that the data file is of the correct length and contains complete records.

OSD-04301

Record size too large

Cause: The specified record size is too large to load.

Action: Reduce record size and reload the data.

OSD-04302

Invalid record type and/or load options

Cause: The control file's Windows NT file processing options string contains an invalid option or keyword.

Action: Set the Windows NT file processing options string to an acceptable value.

Semaphore Errors: OSD-04400 to OSD-04499

OSD-04400

Unable to acquire internal semaphore for process

Cause: Oracle8 database has exceeded the maximum number of connections.

Action: Delete any unused connections and try again.

OSD-04401

WaitForSingleObject() failure, unable to obtain semaphorex

Cause: There was an unexpected return from the Windows NT system service, WaitForSingleObject().

Action: Check the operating system error code and consult the Windows NT documentation.

Miscellaneous Errors: OSD-04500 to OSD-04599

OSD-04500

Illegal option specified

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04501

Internal buffer overflow

Cause: This is an internal error, not normally expected to occur.

Action: Contact Oracle Support Services.

OSD-04502

Translations nested too deep

Cause: The program encountered too many intermediate translations while attempting to translate a configuration variable.

Action: Simplify the values of configuration parameters to include fewer intermediate translations.

OSD-04503

Text contains no translatable elementsx

Cause: The program cannot recognize variables in the text to be translated.

Action: Check and, if necessary, correct the text to be translated.

OSD-04505

Stdin not responding

Cause: The system is unable to receive input from the standard input stream.

Action: Verify that the process has access to an input device.

OSD-04506

Unable to spawn process via system()

Cause: The system is out of memory or the executable is invalid.

Action: Shut down unnecessary processes; install more memory in the computer. Verify the name of the executable.

OSD-04507

Password for 'internal' is incorrect

Cause: An attempt was made to connect as 'internal' with an invalid password.

Action: Verify that the password is correct and try again.

OSD-04508

No password given

Cause: An attempt was made to connect as 'internal' without a password.

Action: Enter a valid password when connecting as internal.

OSD-04509

No password found

Cause: Oracle was unable to locate and retrieve the password for 'internal'.

Action: Verify that Oracle is installed and configured correctly.

OSD-04510

Operating system roles are not supported

Cause: An attempt was made to use an operating system role.

Action: Only use roles that were created 'IDENTIFIED BY *PASSWORD*' as opposed to 'IDENTIFIED EXTERNALLY'.

OSD-04511

Unable to get date and time from the operating system

Cause: There was an unexpected return from GetLocalTime() call.

Action: Verify that the system time is correct on the computer.

OSD-04512

Unable to translate the 'USERNAME' configuration variable on server

Cause: The 'USERNAME' configuration parameter variable on the host is not properly set.

Action: Verify the 'USERNAME' variable is set.

OSD-04513

'REMOTE_OS_AUTHENT' variable not set to TRUE'

Cause: For remote operating system log on to function, the 'REMOTE_OS_AUTHENT' parameter must be set to TRUE.

Action: Shut down and start up the instance with 'REMOTE_OS_AUTHENT = TRUE' in the initialization parameter file.

OSD-04514

The Windows NT Group name is too long for internal buffer

Cause: The Windows NT Group name is too long.

Action: Use a shorter Windows NT group name.

Database Connection Issues

This table lists and provides answers to common Oracle8*i* database connection issues:

If You Receive This Error...	Ensure Your...
`TNS-12203 TNS: unable to connect to destination`	OracleService*SID* and Oracle*Home_ Name*TNSListener[1] services are started. See the Net8 Administrator's Guide.
`ORA-12547 TNS: lost contact`	OracleService*SID* and Oracle*Home_ Name*TNSListener[1] services are started. You receive this error if you attempt to use any of the Oracle8*i* Utilities, such as SQL*Plus. See Managing Oracle Services on page 7-2 for instructions. **Note:** This error is analogous to the following Oracle7 error: `ORA-09352: Windows 32-bit Two-Task driver unable to spawn new ORACLE task`

`ORA-28575: unable to open RPC connection to external procedure agent`
`ORA-06512: at "APPLICATIONS.OSEXEC", line 0`
`ORA-06512: at "APPLICATIONS.TEST", line 4`
`ORA-06512: at line 2`
	TNSNAMES.ORA and LISTENER.ORA files have been correctly configured to use external routines. See Chapter 8 of *Net8 Administrator's Guide*.

[1] Ensure that Oracle*HOME_NAME*TNSListener is started if you are using an Oracle8*i* database that has a home name.

Glossary

alert file

A file that contains important information and error messages that are generated during database operations.

authenticate

To verify the identity of a user, device, or other entity in a computer system, often as a prerequisite for allowing access to resources in a system.

authorization

Permission given to a user, program, or process to access an Oracle database or operating system.

backup

A representative copy of data. This copy includes important parts of your database such as the control file, redo log files, and data files.

A backup is a safeguard against unexpected data loss; if you lose your original data, you can use the backup to make the data available again. A backup is also a safeguard against an application error; if an application makes incorrect changes, you can restore the backup.

connect string

See "net service name".

control file

A file that records the physical structure of a database and contains the database name, the names and locations of associated databases and online redo log files, the timestamp of the database creation, the current log sequence number, and checkpoint information.

Common Object Request Broker Architecture (CORBA)

A standard that enables distributed objects to communicate with each other, independent of programming language, operating system, and location.

data dictionary

A set of read-only tables that provide information about a database.

database alias

See "net service name".

downgrade

To transform an installed version of an Oracle database from a later release back into an earlier release.

Dynamic Link Library (DLL)

An executable file that a Windows application can load when needed.

Enterprise JavaBeans

A server-side component model for Java.

external role

Roles created and managed by the Windows NT operating system. Once an external role is created, you can grant or revoke that role to a database user. You must set the INIT.ORA parameter OS_ROLES to TRUE and restart your Oracle database before you can create an external role. You cannot use both Windows NT and the Oracle database to grant roles concurrently.

external user

A user authenticated by the Windows NT operating system who can access the Oracle database without being prompted for a password. External users are typically regular database users (non-database administrators) to which you assign standard database roles (such as CONNECT and RESOURCE), but do not want to assign SYSDBA (database administrator) or SYSOPER (database operator) privileges.

external routine

A function written in a third-generation language (3GL), such as C, and callable from within PL/SQL or SQL as if it were a PL/SQL function or procedure.

HOME*ID*

Represents a unique registry subkey for each Oracle home directory in which you install products. A new HOME*ID* is created and incremented each time you install products to a different Oracle home directory on one computer. Each HOME*ID* contains its own configuration parameter settings for installed Oracle products.

HOME_NAME

Represents the name of an *ORACLE_HOME*. In release 8.1.6, all Oracle homes have a unique *HOME_NAME*.

initialization parameter file

An ASCII text file that contains information needed to initialize a database and instance.

instance

Every running Oracle database is associated with an Oracle instance. When a database is started on a database server (regardless of the type of computer), Oracle allocates a memory area called the System Global Area (SGA) and starts one or more Oracle processes. This combination of the SGA and the Oracle processes is called an instance. The memory and processes of an instance manage the associated database's data efficiently and serve the one or more users of the database.

instantiate, instantiation

Producing a more defined version of some object by replacing variables with values (or other variables).

In object-oriented programming, producing a particular object from its class template. This involves allocation of a structure with the types specified by the template, and initialization of instance variables with either default values or those provided by the constructor function of the class.

Internet Inter-ORB Protocol (IIOP)

A standard that enables Object Request Brokers (ORBs) from different vendors to communicate with each other using TCP/IP.

listener

The server process that listens for and accepts incoming connection requests from client applications. Oracle listener processes start up Oracle database processes to handle subsequent communications with the client.

LISTENER.ORA

A configuration file that describes one or more Transparent Network Substrate (TNS) listeners on a server.

local role

Roles created and managed by the database. Once a local role is created, you can grant or revoke that role to a database user. You cannot use both Windows NT (for external roles) and the Oracle database (for local roles) to grant roles concurrently.

Microsoft Management Console

An application that serves as a host for administrative tools called snap-ins. By itself, Microsoft Management Console does not provide any functionality.

Microsoft Transaction Server

A COM-based transaction processing system that runs on an Internet or network server.

migrate

To transform an installed version of an Oracle database from a major release to another major release, for example, from Oracle8 to Oracle8*i*.

mount

To associate a database with an instance that has been started.

multiple Oracle homes

The capability of having more than one ORACLE_HOME on a computer.

National Language Support (NLS)

The Oracle architecture that ensures that database utilities, error messages, sort order, date, time, monetary, numeric, and calendar conventions automatically adapt to the native language and locale.

net service name

The name used by clients to identify a Net8 server. A net service name is mapped to a port number and protocol. Also known as a connect string, database alias, or service name.

Net8

The Oracle network interface that enables Oracle tools running on network workstations and servers to access, modify, share, and store data on other servers.

network listener

A listener on a server that listens for connection requests for one or more databases on one or more protocols. See "listener".

network service

In an Oracle application network, a service performs tasks for its service consumers. For example, a Names Server provides name resolution services for clients.

NLS

See "National Language Support (NLS)".

NT global groups

Contains users with access to computers and resources throughout the current domain and within any other domains that trust it. Global groups only contain global domain user accounts as their members.

Object Request Broker (ORB)

A software component that serves as the middleware between distributed objects. The distributed objects must comply with the Common Object Request Broker Architecture (CORBA) standard.

Optimal Flexible Architecture (OFA)

A set of file naming and placement guidelines for Oracle software and databases.

Oracle8*i* Enterprise Edition and Oracle8*i*

The information in this guide applies to both the Oracle8*i* Enterprise Edition and Oracle8*i* database types. Unless otherwise noted, the features and functionality described in this guide are common to both Oracle8*i* Enterprise Edition and Oracle8*i*.

Oracle Call Interface (OCI)

An application programming interface that enables you to manipulate data and schemas in an Oracle database. You compile and link an Oracle Call Interface program in the same way that you compile and link a non-database application. There is no need for a separate preprocessing or precompilation step.

ORACLE_HOME

Corresponds to the environment in which Oracle products run. This environment includes the location of installed product files, the *PATH* variable pointing to the products' binary files, registry entries, net service names, and program groups.

If you install an OFA-compliant database, using Oracle Universal Installer defaults, Oracle home (known as *\ORACLE_HOME* in this guide) is located beneath *X:\ORACLE_BASE.* It contains subdirectories for Oracle software executables and network files.

Oracle JServer or Oracle JServer Enterprise Edition

Oracle8*i* includes Oracle JServer, the integrated Java Virtual Machine. Oracle JServer provides Java2 support (JDK1.2), a CORBA 2.0 Object Request Broker, an embedded JDBC driver, a SQLJ translator, and an Enterprise JavaBeans transaction server.

Oracle Protocol Support

A product that maps the functions of a given network protocol into Oracle Transparent Network Substrate (TNS) architecture. This process translates TNS function calls into requests to the underlying network protocol. This allows TNS to act as an interface among all protocols. Net8 requires Oracle protocol support.

Oracle service

A service that is associated with an Oracle component.

ORACLE_BASE

Oracle base, known as *ORACLE_BASE* in this guide, is the root of the Oracle directory tree.

If you install an OFA-compliant database using Oracle Universal Installer defaults, *ORACLE_BASE* is *X*:\ORACLE where *X* is any hard drive (for example, C:\ORACLE).

PL/SQL

Oracle Corporation's procedural language extension to SQL.

PL/SQL enables you to mix SQL statements with procedural constructs. You can define and execute PL/SQL program units such as procedures, functions, and packages.

precompiler

A programming tool that enables you to embed SQL statements in a high-level source program.

privilege

A right to execute a particular type of SQL statement or to access another user's object.

process

A mechanism in an operating system that can run an executable. (Some operating systems use the terms job or task.) A process normally has its own private memory area in which it runs. On Windows NT a process is created when a program runs (such as Oracle or Microsoft Word). In addition to an executable program, all processes consist of at least one *thread*. The ORACLE master process contains hundreds of threads.

quota

A limit on a resource, such as a limit on the amount of database storage used by a database user. A database administrator can set tablespace quotas for each Oracle user name.

raw partition

A portion of a physical disk that is accessed at the lowest possible disk (block) level.

recovery

To *restore* a physical backup is to reconstruct it and make it available to the Oracle server. To *recover* a restored backup is to update it using redo records (that is, records of changes made to the database after the backup was taken). Recovering a backup involves two distinct operations: rolling forward the backup to a more current time by applying redo data, and rolling back all changes made in uncommitted transactions to their original state.

redo log file

A file that contains a record of all changes made to data in the database buffer cache. If an instance failure occurs, the redo log files are used to recover the modified data that was in memory.

redo log buffer

A circular buffer in the System Global Area (SGA) that contains information about changes made to the database.

registry

A Windows repository that stores configuration information for a computer.

remote computer

A computer on a network other than the local computer.

remote database

A database on a computer other than the local database.

replication

The process of copying and maintaining database objects in multiple databases that make up a distributed database system.

role

A named group of related privileges. You can grant a role to users or other roles.

schema

A named collection of objects, such as tables, views, clusters, procedures, and packages, associated with a particular user.

service

An executable process installed in the Windows NT registry and administered by Windows NT. Once a service is created and started, it can run even when no user is logged on to the computer.

service name

See "net service name".

SID

See "system identifier (SID)".

snap-in

An administrative tool that runs within Microsoft Management Console.

snapshot

(1) Information stored in rollback segments to provide transaction recovery and read consistency. Rollback segment information can be used to recreate a snapshot of a row before an update.

(2) A read-only copy of a master table located on a remote node. Snapshots can be queried, but not updated; only the master table can be updated. Snapshots are periodically refreshed to reflect changes made to the master table.

starter database

A preconfigured, ready-to-use database that requires minimal user input to create.

synonym

An alias for a table, view, sequence, or program unit. A synonym is not actually an object itself; rather, it is a direct reference to its base object.

SYSDBA

A special database administration role that contains all system privileges with the ADMIN OPTION, and the SYSOPER system privilege. SYSDBA also permits CREATE DATABASE actions and time-based recovery.

SYSOPER

A special database administration role that permits a database administrator to perform STARTUP, SHUTDOWN, ALTER DATABASE OPEN/MOUNT, ALTER DATABASE BACKUP, ARCHIVE LOG, and RECOVER, and includes the RESTRICTED SESSION privilege.

System Global Area (SGA)

A group of shared memory structures that contain data and control information for an Oracle instance.

system identifier (SID)

A unique name for an Oracle instance. To switch between Oracle databases, users must specify the desired SID. The SID is included in the CONNECT DATA parts of the connect descriptors in a TNSNAMES.ORA file, and in the definition of the network listener in a LISTENER.ORA file.

SYSTEM user name

One of two standard DBA user names automatically created with each database. (The other user name is SYS.) SYSTEM is created with an initial password of MANAGER. The SYSTEM user name is the preferred user name for DBAs to use for database maintenance.

tablespace

A database is divided into one or more logical storage units called tablespaces. Tablespaces are divided into logical units of storage called segments, which are further divided into extents.

thread

An individual path of execution within a process. Threads are objects within a process that execute program instructions. Threads allow concurrent operations within a process so that a process can execute different parts of its program simultaneously on different processors. A thread is the most fundamental component that can be scheduled on Windows NT.

TNSNAMES.ORA

A file that contains connect descriptors mapped to net service names. The file may be maintained centrally or locally, for use by all or individual clients.

trace file

Each server and background process can write to an associated trace file. When a process detects an internal error, it dumps information about the error to its trace file. Some of the information written to a trace file is intended for the database administrator, while other information is intended for Oracle Support Services. Trace file information is also used to tune applications and instances.

upgrade

To transform an installed version of an Oracle database major release into another major release of the same version. Compare with "migrate".

user name

A name that can connect to and access objects in a database.

view

A selective presentation of the structure of, and data in, one or more tables (or other views).

Index

Symbols

"", SQL*Loader parameter,   2-18
"FIX n", SQL*Loader parameter,   2-18
"RECSIZE", SQL*Loader parameter,   2-18
"VAR xxxx", SQL*Loader parameter,   2-19
\\.\PhysicalDrive*x*,   D-4

Numerics

1521 port,   E-11
1526 port,   E-11
1575 port,   E-11
1610 port,   E-11
1620 port,   E-11
4GB RAM Tuning,   10-3

A

Access Control Lists
    accessing security groups,   4-21
    adding or removing users in security groups,   4-23
    available security groups,   4-20
    changing user permissions in security groups,   4-24
Active Directory
    accessing security groups,   4-21
    adding or removing users in security groups,   4-23
    automatic discovery of directory servers,   4-6
    changing permissions in security groups,   4-24
    connecting to a database,   4-16, 4-17
    creating security domains,   4-25
    defined,   4-2
    enhancement of directory object type descriptions,   4-8
    how Oracle directory objects appear,   4-9
    installation requirements,   8-10
    integration with directory servers,   4-6
    integration with Microsoft tools,   4-7
    integration with Windows login credentials,   4-8
    managing Access Control Lists,   4-20
    managing security groups directory server managing Access Control Lists,   4-20
    procedures for installing Oracle into,   4-14
    required Oracle configuration tools,   4-14
    requirements for creating an Oracle Context,   4-11
    requirements for creating an Oracle schema,   4-10
    requirements for creating enterprise user security,   4-13
    requirements for creating Net8 directory naming,   4-12
    requirements for using Oracle,   4-10
    testing connectivity from client computers,   4-16
    testing connectivity with Microsoft tools,   4-17
    testing connectivity with SQL*Plus,   4-7, 4-19
    testing database connectivity,   4-7, 4-18
    user interface extensions,   4-7
Active Directory Users and Computers
    accessing directory server objects,   4-17
    integration with Oracle objects in Active Directory,   4-7
    testing database connectivity from,   4-18
ADMIN directory, explained,   3-22
administration tools
    for enterprise users,   8-9
    for external users,   8-6

S

X

X.500 naming convention,   4-16
XAVIEW.SQL script
   location of,   A-5