

Oracle8i Parallel Server

Setup and Configuration Guide

Release 8.1.5

February 1999

Part No. A67439-01

Part No. A67439-01

Copyright © 1999, Oracle Corporation. All rights reserved.

Primary Author: Deborah Steiner

Contributors: James Cassidy, Jonathon Creighton, Ivan Lam, Jimmy Yu, Peter Sciarra, Trevor Selby, and Tak Wang

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and Oracle disclaims liability for any damages caused by such use of the Programs.

The Programs (which include both the software and documentation) contain proprietary information of Oracle Corporation; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Oracle Corporation does not warrant that this document is error free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Oracle Corporation.

If the Programs are delivered to the U.S. Government or anyone licensing or using the Programs on behalf of the U.S. Government, the following notice is applicable:

Restricted Rights Notice Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

Oracle, SQL*Net and SQL*Plus are registered trademarks of Oracle Corporation. Oracle8i and Net8 are trademarks of Oracle Corporation. All other company or product names mentioned are used for identification purposes only and may be trademarks of their respective owners.

Contents

Send Us Your Comments	xi
Preface.....	xiii
1 Introducing Oracle Parallel Server	
Introduction	1-2
Oracle Parallel Server and the Optimal Flexible Architecture	1-2
Oracle Parallel Server Software Components.....	1-3
Operating System Dependent Layer	1-4
Oracle Parallel Server Instances	1-9
Unique Identification of Instances	1-11
Data Files.....	1-14
Oracle Parallel Server Features	1-15
Integrated Distributed Lock Manager	1-15
Parallel Cache Management.....	1-17
Oracle Parallel Query.....	1-20
Oracle Parallel Query Statement	1-22
2 Pre-Installation	
System Installation Requirements	2-2
Hardware and Software Requirements for Oracle8i Parallel Server Node	2-2
Hardware and Software Requirement for Oracle Enterprise Manager.....	2-3
Shared Disk Subsystem	2-5
Setting Up Raw Devices	2-6
Identifying Raw Devices	2-10
Pre-Installation Steps.....	2-13

3	Installation	
	Installation Types.....	3-2
	Installation Procedure.....	3-3
	Migrating or Upgrading to Release 8.1.....	3-8
	Multiple Oracle Homes.....	3-9
4	Configuring Oracle Parallel Server	
	Configuration Options.....	4-2
	Creating the Database.....	4-3
	Step 1: Create and Identify Raw Partitions.....	4-3
	Step 2: Create the Database.....	4-4
	Starting the Database in Parallel Mode.....	4-9
	Verifying Instances Are Running.....	4-11
	Configuring Net8 for Nodes.....	4-12
	Step 1: Configure the Listener.....	4-13
	Step 2: Configure Net Service Names.....	4-20
	Step 3: Configure SQLNET.ORA.....	4-32
	Configuring Net8 for Clients.....	4-34
	Configuring Net8 with Failover.....	4-35
	Configure the Listener.....	4-36
	Configuring the TNSNAMES.ORA File for Failover.....	4-39
	Testing the Net8 Configuration.....	4-45
5	Installing and Configuring Oracle Parallel Server Management	
	Oracle Parallel Server Management Architecture.....	5-2
	Oracle Parallel Server Requirements.....	5-7
	Understanding Oracle Enterprise Manager Setup.....	5-8
	Installing Oracle Enterprise Manager.....	5-10
	Configuring Oracle Enterprise Manager.....	5-11
	Step 1: Configure Net8.....	5-11
	Step 2: Configure OPSCTL for UNIX Platforms.....	5-12
	Step 3: Configure Repository.....	5-13
	Step 4: Start Oracle Enterprise Manager Components.....	5-13
	Step 5: Discover Nodes.....	5-18
	Step 6: Specify Preferred Credentials for Nodes and Oracle Parallel Server Database....	5-21

Installing Oracle Performance Manager	5-28
Configuring Oracle Performance Manager.....	5-29
Step 1: Configure Net8.....	5-29
Step 2: Start Oracle Performance Manager Components	5-30
Step 3: Accessing Oracle Parallel Server Charts.....	5-34

6 Administering Oracle Parallel Server

Understanding the Initialization Files	6-2
Purpose of INITSID.ORA.....	6-3
Purpose of INITDB_NAME.ORA.....	6-5
Setting Initialization Parameters.....	6-8
GC_* Global Constant Parameters.....	6-8
Parameter Notes for Multiple Instances.....	6-10
Identical Parameters on Each Instance.....	6-11
Setting LM_* Parameters.....	6-12
Parameter Descriptions.....	6-12
Location of Initialization Files	6-12
Editing Initialization Files	6-12
Managing Instances Using Oracle Parallel Server Management.....	6-13
Starting the Console	6-14
Displaying Oracle Parallel Server Objects in the Navigator Window.....	6-14
Using the Right-Mouse Menu.....	6-17
Starting an Oracle Parallel Server Database	6-18
Shutting Down an Oracle Parallel Server Database	6-21
Viewing the Parallel Server Operation Results.....	6-24
Viewing Oracle Parallel Server Status.....	6-27
Creating a Job for a Parallel Server or Parallel Server Instance.....	6-31
Specifying Job Details	6-32

7 Monitoring Performance with Oracle Parallel Server Management Charts

Oracle Performance Manager Overview	7-2
Starting the Oracle Performance Manager	7-4
Displaying Charts	7-4
Parallel Server Block Ping Chart.....	7-6
Parallel Server Data Block Ping by Tablespace Chart	7-7
Parallel Server Data Block Ping by Instance Chart	7-9
Parallel Server File I/O Rate Chart	7-10
Parallel Server File I/O Rate by Object Chart	7-11
Parallel Server File I/O Rate by Instance Chart	7-12
Parallel Server Lock Activity Chart.....	7-13
Parallel Server Sessions Chart.....	7-14
Parallel Server Users Logged On Chart.....	7-15
Parallel Server Users Logged On by Instance Chart	7-16
Parallel Server Active Users Chart	7-17
Parallel Server Active Users by Instance Chart.....	7-18
OPS Overview Chart.....	7-19

8 Adding Instances and Nodes

Moving from a Single Instance to Parallel Server	8-2
Step 1: Configure Hardware	8-2
Step 2: Evaluate Tablespaces and Log Files of Single Instance	8-2
Step 3: Create Raw Devices	8-3
Step 4: Export Data from Old Database.....	8-3
Step 5: Install Operating System Dependent Files	8-3
Step 6: Install Oracle8i Enterprise Edition and Oracle8i Parallel Server Option.....	8-4
Step 7: Create the Database	8-4
Step 8: Transfer Data From Old Data to Empty Database	8-5
Step 9: Start the Database.....	8-5
Adding Additional Nodes to a Cluster	8-6

A Directory Structure

Understanding the Oracle Parallel Server Directory Structure	A-2
UNIX.....	A-2
Windows NT.....	A-3

B Oracle Parallel Server Management on UNIX Reference

OPSQL Utility	B-2
Creating the <i>DB_NAME.CONF</i> File	B-2
Parameter Syntax.....	B-3
Parameter Expansion	B-3
Parameter Descriptions.....	B-5
Parameter Node Overrides	B-7
Oracle Parallel Server Instances	B-7
Listeners	B-9
Sample of <i>DB_NAME.CONF</i> File.....	B-11
Oracle Intelligent Agent Configuration	B-11
Non-shared Oracle Home	B-12
Shared Oracle Home	B-12

C Troubleshooting

Cleaning Up the Registry After an Oracle Database Configuration Assistant Failure on Windows NT	C-2
Handling Auto-Discovery Failures	C-3
Understanding Auto-Discovery	C-3
Auto-Discovery Results.....	C-9
Troubleshooting Auto-Discovery.....	C-10
Using Trace Files	C-11
Background Thread Trace Files.....	C-11
User Thread Trace Files	C-12
Alert File.....	C-12
Error Call Trace Stack	C-13
Contacting Oracle Worldwide Customer Support	C-14
Severe Errors	C-14

Glossary

Index

List of Figures

1-1	Operating System Dependent Components.....	1-5
1-2	Oracle Parallel Server Instance Architecture.....	1-14
1-3	Parallel Query Operation	1-21
5-1	OPSQL Architecture for UNIX.....	5-5
5-2	OPSQL Architecture for Windows NT	5-6
5-3	Console, Management Server and Repository on the Same Machine.....	5-8
5-4	Console, Management Server and Repository on the Separate Machines.....	5-9
6-1	Instance Initialization Files	6-2
6-2	INITOP1.ORA	6-3
6-3	INITOP2.ORA	6-3
6-4	Common Initialization Files.....	6-5
6-5	INITOP.ORA.....	6-6
7-1	Parallel Server Block Ping Chart	7-6
7-2	Parallel Server Block Ping by Tablespace Chart	7-7
7-3	OPS Block Ping Tablespace Object Chart	7-8
7-4	OPS Block Ping Tablespace Instance Chart	7-8
7-5	Parallel Server Block Ping by Instance Chart	7-9
7-6	Parallel Server File IO Rate Chart	7-10
7-7	Parallel Server Block Ping by Object Chart	7-11
7-8	Parallel Server File IO Rate by Instance Chart	7-12
7-9	Parallel Server Lock Activity Chart	7-13
7-10	Parallel Server Sessions Chart	7-14
7-11	Parallel Server Users Logged On Chart	7-15
7-12	Parallel Server Users Logged On by Instance Chart	7-16
7-13	Parallel Server Active Users Chart.....	7-17
7-14	Parallel Server Active Users by Instance Chart	7-18
7-15	Overview Chart	7-19

Send Us Your Comments

Oracle8i Parallel Server Setup and Configuration Guide, Release 8.1.5
Part No. A67439-01

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 manual?

If you find any errors or have any other suggestions for improvement, please indicate the chapter, section, and page number (if available). You can send comments to the Information Development department in the following ways:

- Electronic mail - infodev@us.oracle.com
- FAX - (650) 506-7228 Attn: Oracle Server Documentation
- Postal service:
Oracle Corporation
Server Documentation Manager
500 Oracle Parkway, 4OP12
Redwood Shores, CA 94065
USA

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

If you have problems with the software, please contact your local Oracle World Wide Support Center.

Preface

The *Oracle8i Parallel Server Setup and Configuration* guide provides the information you need to install, configure and manage Oracle Parallel Server in an Oracle8i environment.

This preface includes the following topics:

- [Intended Audience](#)
- [Prerequisites](#)
- [Structure](#)
- [Related Documents](#)
- [Documentation and Code Conventions](#)
- [Your Comments Are Welcome](#)

Intended Audience

The information in this guide is intended primarily for network or database administrators (DBAs) responsible installation and configuration of Oracle Parallel Server.

Prerequisites

This guide assumes that you are familiar with:

- the target operating system and have installed and tested it for the cluster and network hardware
- object relational database management concepts

Structure

This guide contains eight chapters and three appendices.

Chapter 1, "Introducing Oracle Parallel Server"

Describes the main features and components of Oracle Parallel Server.

Chapter 2, "Pre-Installation"

Describes the specific hardware and software, raw device, and pre-installation requirements for installing Oracle Parallel Server.

Chapter 3, "Installation"

Describes how to install the components needed for Oracle Parallel Server.

Chapter 4, "Configuring Oracle Parallel Server"

Describes how to configure and start an Oracle Parallel Server database, as well as configure networking components.

Chapter 5, "Installing and Configuring Oracle Parallel Server Management"

Describes the tasks to install and configure Oracle Enterprise Manager and Oracle Performance Manager for parallel server management.

Chapter 6, "Administering Oracle Parallel Server"

Describes Oracle Parallel Server initialization files and administration of Oracle Parallel Server with the Oracle Enterprise Manager Console.

Chapter 7, "Monitoring Performance with Oracle Parallel Server Management Charts"

Describes how to use Oracle Performance Manager to generate metric charts for Oracle Parallel Server databases.

Chapter 8, "Adding Instances and Nodes"

Describes how to add nodes to an existing cluster, and how to migrate from an single-instance database to multiple instances.

Appendix A, "Directory Structure"

Describes the directory structure for Oracle Parallel Server.

Appendix B, "Oracle Parallel Server Management on UNIX Reference"

Provides reference information for Oracle Parallel Server Management on UNIX.

Appendix C, "Troubleshooting"

Describes how to resolve common Oracle Parallel Server configuration problems.

Glossary

Provides brief descriptions of terms used throughout this guide.

Related Documents

For more information, see the following user guides:

- *Oracle8i Administrator's Reference* for Sun Solaris, HP 9000 or AIX-based systems
- *Oracle8i Installation Guide* for Sun Solaris, HP 9000 or AIX-based systems
- *Oracle8i Enterprise Edition Installation for Windows NT*
- *Oracle8i Parallel Server Getting Started for Windows NT*
- *Oracle8i Concepts*
- *Net8 Administrator's Guide*
- *Getting to Know Oracle8i*
- *Oracle8i Parallel Server Concepts and Administration*
- *Oracle8i Reference*

- *Oracle Enterprise Manager Installation*
- *Oracle Enterprise Manager Configuration Guide*
- *Oracle Enterprise Manager Administrator's Guide*
- *Oracle Diagnostics Installation*
- *Oracle Enterprise Manager Performance Monitoring and Planning Guide*

Documentation and Code Conventions

The following conventions are used in this guide:

Convention	Example	Meaning
UPPERCASE	LISTENER.ORA	<p>Calls attention to SQL commands, keywords, file names, and initialization parameters.</p> <p>Most Oracle files names on UNIX are lowercase.</p>
lowercase courier	\$ORACLE_HOME/network/admin startup;	<p>Indicates directories, commands and case-sensitive parameters.</p> <p>Most directories and commands on UNIX are lowercase.</p>
Italic	<p>Italic is used to indicate a variable: <i>filename</i></p> <p>Italic is used for emphasis: The WHERE clause may be used to <i>join</i> rows in different tables.</p>	<p>Indicates a value that you must provide. For example, if a command asks you to type <i>filename</i>, you must type the actual name of the file.</p> <p>Italic is also used for emphasis in the text and to indicate the titles of other guides.</p>
Choose Start >	Choose Start > Programs > Oracle- <i>HOME_NAME</i> > Oracle Enterprise Management > Enterprise Manager Console	Indicates how to start a program on Windows platforms.
Forward slash (/) before a directory name	/network	A subdirectory on UNIX platforms.
Backslash (\) before a directory name	\network	A subdirectory on Windows platforms.

Convention	Example	Meaning
\$ORACLE_HOME on UNIX and <i>ORACLE_HOME</i> on Windows platforms	Go to the \$ORACLE_HOME\network directory	In this Optimal Flexible Architecture (OFA)-compliant release, all subdirectories are no longer under a top level ORACLE_HOME directory. There can be an ORACLE_BASE directory, which may contain multiple Oracle home directories.
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.

Your Comments Are Welcome

We value and appreciate your comments as an Oracle user and reader of the user guides. As we write, revise, and evaluate our documentation, your opinions are the most important input we receive. Please use the reader's comment form to tell us what you like or dislike about this manual or other Oracle manuals. If the form is not available, please send your comments to the Information Development department using any of the following:

- E-mail: infodev@us.oracle.com
- FAX - (650) 506-7228. Attn: Oracle Server Documentation
- Postal Service:

Oracle Corporation
 Server Documentation Manager
 500 Oracle Parkway, 4OP12
 Redwood Shores, CA 94065
 U.S.A.

Introducing Oracle Parallel Server

This chapter provides a conceptual and component overview of Oracle Parallel Server. This information helps you prepare and plan your Oracle Parallel Server installation and configuration in an Oracle8i environment.

Specific topics discussed are:

- [Introduction](#)
- [Oracle Parallel Server and the Optimal Flexible Architecture](#)
- [Oracle Parallel Server Software Components](#)
- [Oracle Parallel Server Instances](#)
- [Oracle Parallel Server Features](#)

Introduction

Oracle Parallel Server is an architecture that allows multiple instances to access a shared database. Oracle Parallel Server offers the following (terms will be described later in this chapter):

- Superior scalability through distribution of workload across nodes
- High-availability through multiple nodes accessing the database; if one node fails, the database is still accessible through surviving nodes

An Oracle Parallel Server can handle node or disk failure in a clustered environment with no or minimal downtime. The Oracle Parallel Server architecture provides the following features:

- Separate Oracle instances running on different nodes operate against a set of common data files that reside on shared disks physically accessible by all nodes that make up the cluster.
- All instances can execute transactions concurrently against the same database, and each instance can have multiple users executing transactions concurrently.

Coordination of each node accessing the shared database provides the following:

- Data consistency and integrity
- Data and lock recovery support

Oracle Parallel Server and the Optimal Flexible Architecture

This guide describes the features of the Oracle Parallel Server in relation to the Optimal Flexible Architecture (OFA), whereby Oracle software and database directories and file names following a particular format. This guide recommends using tools that adhere to the OFA structure, such as the Oracle Database Configuration Assistant, to configure and administer an Oracle Parallel Server databases. Therefore, all examples and features are described from a OFA point-of-view. You are free to use other naming conventions.

Oracle Parallel Server Software Components

The following components comprise Oracle Parallel Server:

Component	Description
Oracle8i Enterprise Edition	Provides the applications and files to manage a database. All other Oracle Parallel Server components are layered on top of Oracle8i Enterprise Edition.
Oracle Parallel Server Option	Provides the necessary Oracle Parallel Server scripts, initialization files, and data files.
Oracle Parallel Server Management	Provides a single point for starting, stopping, and monitoring the activity of parallel servers and parallel server instances from within Oracle Enterprise Manager Console. See Chapter 5, "Installing and Configuring Oracle Parallel Server Management" and Chapter 6, "Administering Oracle Parallel Server" .
Operating System Dependent layer	Consists of several software components developed by vendors. The Operating System Dependent layer maps the key OS/cluster-ware services required for proper operation of Oracle Parallel Server. See "Operating System Dependent Layer" on page 1-4.

Operating System Dependent Layer

A vendor-supplied Operating System Dependent layer that passed certification must be installed after Oracle Parallel Server Option is installed. The Operating System Dependent layer consists of several software components developed by vendors. The Operating System Dependent layer maps the key OS/cluster-ware services required for proper operation of Oracle Parallel Server.

Operating System Dependent Components

The Operating System Dependent layer consists of:

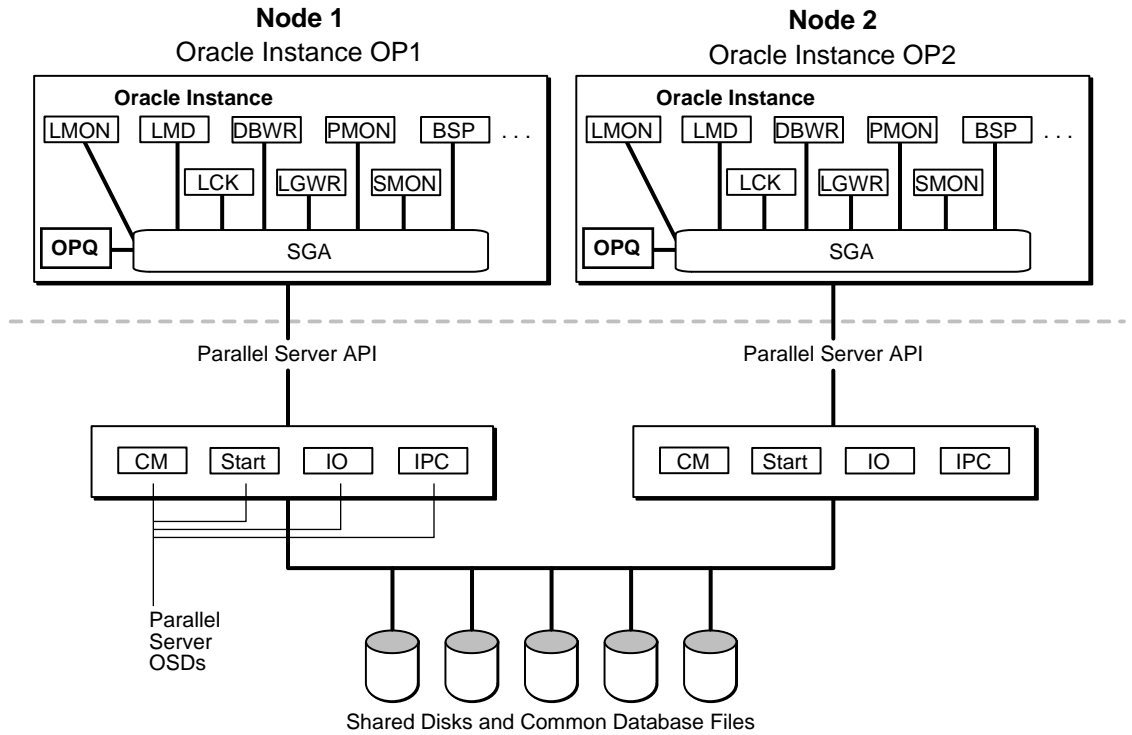
Component	Description	Required?
Cluster Manager (CM)	Discovers the state of the cluster and instance membership.	Yes
Inter-Process Communication (IPC)	Provides reliable transfer of messages between instances on different nodes.	Yes
Input/Output (IO)	Provides I/O to access shared disks.	No
Startup (START)	Provides one-time configuration to startup functionality.	No

Note: IO and START are available on Windows NT only.

These components provide key services required for proper operation of the Oracle Parallel Server Option and are used by various clients, such as Integrated Distributed Lock Manager.

Figure 1-1 illustrates the Operating System Dependent components in a cluster with two nodes:

Figure 1-1 Operating System Dependent Components



Cluster Manager

The Cluster Manager (CM) monitors process health, specifically of the database instance. LMON, a background process that monitors the health of the Integrated Distributed Lock Manager, registers and de-registers from CM.

It is critical that all Oracle Parallel Server instances receive the same membership information when events occur. Notification changes cause relevant Oracle Parallel Server recovery operations to be initiated. If any node or instance is determined to be dead or otherwise not a properly functioning part of the system, CM terminates all processes on that node or instance. Thus, any process or thread running Oracle code can safely assume its node or instance is an active member of the system.

If there is a failure, recovery is transparent to user applications. CM automatically reconfigures the system to isolate the failed node and instance and notify Integrated Distributed Lock Manager of the status. Integrated Distributed Lock Manager subsequently recovers any of the locks from the failed node or instance. Oracle Parallel Server can then recover the database to a valid state.

The Integrated Distributed Lock Manager relies on the Cluster Manager for timely and correct information. If the Integrated Distributed Lock Manager cannot get the information it needs, it will shut down the instance.

CM performs the following two types of operations:

- [Node Monitoring](#)
- [Cluster Management](#)

The description of these operations is based on Oracle's implementation of the Operating System Dependent layer. Your vendor's implementation may differ.

Node Monitoring CM manages access to shared disks and monitors the status of various cluster resources, including nodes, networks and instances.

Node monitoring:

- Provides the basic node management interface modules needed by Oracle Parallel Server in a cluster environment.
- Allows the Oracle Parallel Server environment to discover and track the membership state of nodes by providing a common view of cluster membership across the cluster.
- Runs on all nodes and monitors the topology of the cluster by querying all nodes for their current membership.

- Detects changes in the state of active nodes signaling those events, diagnosing the changes, and coordinating a new common and consistent state among all nodes.
- Notifies Oracle Parallel Server of the actual changes to membership.

Cluster Management

CM determines what groups are up and manages instance members. Each instance registers with its database specific group. This CM function was performed by the Group Membership Service (GMS) prior to this release.

Inter-Process Communication (Node-to-Node)

Oracle Parallel Server derives most of its functional benefits from its ability to run on multiple interconnected machines. Oracle Parallel Server relies heavily on the underlying Inter-Process Communication (IPC) component to facilitate this.

IPC defines the protocols and interfaces required for the Oracle Parallel Server environment to transfer reliable messages between instances. Messages are the fundamental logical units of communication in this interface. The core IPC functionality is built around an asynchronous, queued messaging model. IPC is designed to send/receive discrete messages as fast as the hardware allows. With an optimized communication layer, various services can be implemented above it. This is how the Integrated Distributed Lock Manager carries out all of its communication.

Input/Output

Note: Input/Output is available on Windows NT only.

The Input/Output (IO) component provides interprocess capabilities that a cluster implementation must support to enable proper operation of the Oracle Parallel Server environment.

The Oracle Parallel Server environment is extremely dependent on the ability of the underlying OS/cluster implementation to support simultaneous disk sharing across all nodes that run coordinated Oracle Parallel Server instances. Unlike switch-over based technologies, all Oracle Parallel Server instances are active and can operate on any database entity in the shared physical database simultaneously. It is this capability that gives Oracle Parallel Server a large portion of its parallel scalability.

It is the role of the Integrated Distributed Lock Manager to coordinate the simultaneous access to shared databases in a way that maintains consistency and data integrity.

At a high level, the Oracle Parallel Server shared I/O model can be described as a distributed disk cache implemented across all nodes that define the Oracle Parallel Server cluster. The core of Oracle Parallel Server can be viewed as a major client of the cache. Disk blocks from the shared devices are read into a particular node instance cache only after mediation by the Integrated Distributed Lock Manager. The other node instance may read the same blocks into its cache and operate on them simultaneously. Updates to those blocks are carefully coordinated. In general, all shared disk based I/O operations are mediated by the Integrated Distributed Lock Manager. The set of distributed Integrated Distributed Lock Managers on each node can be thought of as managing the distributed aspects of the cache.

Disk update operations must be carefully coordinated so that all nodes see the same data in a consistent way. Any Oracle Parallel Server instance intending to update a cached data block must enter into a dialog with the Integrated Distributed Lock Manager to ensure it has exclusive right to update the block. Once it does this, the instance is free to update the block until its rights have been revoked by the Integrated Distributed Lock Manager. When the exclusive update right is revoked, the instance with block updates must write the block to disk so that the other node can see the changes. Given this rather high-level view of the Integrated Distributed Lock Manager I/O consistency model, it is clear that disk blocks can migrate around to each instance's block cache and all updates are flushed to disk when an instance other than the owner desires access to the block. It is this property that directly determines the reliance of Oracle Parallel Server on shared disk implementations.

Startup

Note: Startup is available on Windows NT only.

The Startup (START) component initiates the Oracle Parallel Server components in a specific order during instance startup. It is up to the vendor to determine this startup sequence.

Oracle Parallel Server Instances

Each Oracle Parallel Server *instance*, a running instance of Oracle8i software, is comprised of a System Global Area (SGA) and Oracle background processes. The SGA is a shared memory region that contains data and control information for an instance. Users can connect to any instance to access the information that resides within the shared database.

Oracle Parallel Server instances coordinate with the following components:

Component	Description
node	A server where an instance resides.
cluster	A set of physically interconnected nodes, and a shared disk storage subsystem.
database	<p>The set of all instances with access to the same data files. A database is limited to a set of Oracle Parallel Server instances that run on only the nodes defined within a cluster.</p> <p>The database name is defined by <i>global database name</i>, a name comprised of the database name and domain, entered during installation.</p>

All instances share:

- the same set of data files
- the same control files

An instance contains:

- at least the following background processes:
 - PMON
 - SMON
 - DBWR
 - LGWR
- additional background processes:
 - LCKN to manage the locks used by an instance and to coordinate requests for those locks by other instances. Additional lock processes, LCK1 through LCK9, are available for systems that require exceptionally high throughput of instance lock requests. The single lock process per instance, LCK0, is usually sufficient for most systems.
 - LMON to monitor the health of the Integrated Distributed Lock Manager.
 - LMD0 to process lock requests
 - BSP0 to manage out-going messages to requesting nodes for the Consistent Read (CR) Server. (The LCK process manages in-coming lock requests for instances holding locks needed by other instances.)
- the Integrated Distributed Lock Manager (IDLM) component in the SGA, an area for global locks and resources. This area was formerly part of the external Distributed Lock Manager (DLM).

See *Oracle8i Concepts* for information about Oracle8i database processes and memory structures.

The *data files* are located on disk drives that are shared between the multiple nodes. If one node fails, client applications (written to do so) can re-route users to another node. One of the surviving nodes automatically performs recovery by rolling back any incomplete transactions that the other node was attempting. This ensures the logical consistency of the database.

Note: Because an instance does not include data files, you can start up an instance without mounting the data files.

Unique Identification of Instances

Following the OFA naming convention, each node's instance has its own *INITSID.ORA* file that uniquely defines the instance with instance-related parameters. This file calls *INITDB_NAME.ORA*, which lists common parameters shared from node-to-node. Both files are created by the Oracle Database Configuration Assistant after installation.

The *INITSID.ORA* file is impacted by a number of elements, including:

Element	Description
Oracle System Identifier (SID)	<p>Uniquely identifies a node's instance.</p> <p>The database name, specified by the <i>DB_NAME</i> parameter in the <i>INITDB_NAME.ORA</i> file, and unique thread ID make up each node's SID. The thread ID starts at 1 for the first instance in the cluster, and is incremented by 1 for the next instance, and so on.</p> <p>The SID is defined as an <i>ORACLE_SID</i> environment variable on UNIX platforms and in the <i>ORACLE_SID</i> registry entry on Windows NT.</p>
Instance Name	<p>Represents the name of the instance and is used to uniquely identify a specific instance when multiple instances share common service names.</p> <p>The instance name is the same as the SID. It is depicted by the <i>INSTANCE_NAME</i> parameter in the <i>INITSID.ORA</i> file.</p>

Element	Description
Rollback Segments	<p>Specifies at least two rollback segments to store "undo" information. If the <code>ROLLBACK_SEGMENTS</code> parameter is set in the <code>INITSID.ORA</code> file, an instance acquires all of the rollback segments named in this parameter, even if the number of rollback segments exceeds the minimum number required by the instance.</p> <p>By default, the Oracle Database Configuration Assistant creates rollback segment identified by <code>RBS$thread_id$_rollback_number</code>. For example, on node with a sequence number of 1, the rollback segments are <code>RBS1-1</code> and <code>RBS1-2</code>.</p>
Thread ID	<p>Specifies the redo thread that is to be used by the instance. Any available redo thread number can be used, but an instance cannot use the same thread number as another instance. Also, an instance cannot start when its redo thread is disabled. An instance cannot mount a database if the thread is used by another instance or if the thread is disabled.</p> <p>The thread starts at 1 node for the first instance in the cluster, and is incremented by 1 for the next instance, and so on. Redo thread numbers are used in the naming of <i>redo log files</i>, which record changes made to data. When redo log files are generated, they include the thread, allowing you to easily identify a particular node's redo log files.</p> <p>Threads are depicted by the <code>THREAD</code> parameter in the <code>INITSID.ORA</code> file.</p>

Examples of Relationships Between Elements

If the `DB_NAME` is `OP` and the thread IDs are 1, 2, and 3, then the SID for each node is:

Thread ID	SID
1	OP1
2	OP2
3	OP3

If the DB_NAME is OP and the node numbers are 0, 1, 4, and 6, then the instance elements are effected in the following manner:

Computer	Thread ID	SID	INSTANCE_NAME	ROLLBACK_ SEGMENT name
Computer A	1	OP1	OP1	RBS1-1, RBS1-2
Computer B	2	OP2	OP2	RBS2-1, RBS2-2
Computer C	3	OP3	OP3	RBS3-1, RBS3-2
Computer D	4	OP4	OP4	RBS4-1, RBS4-2

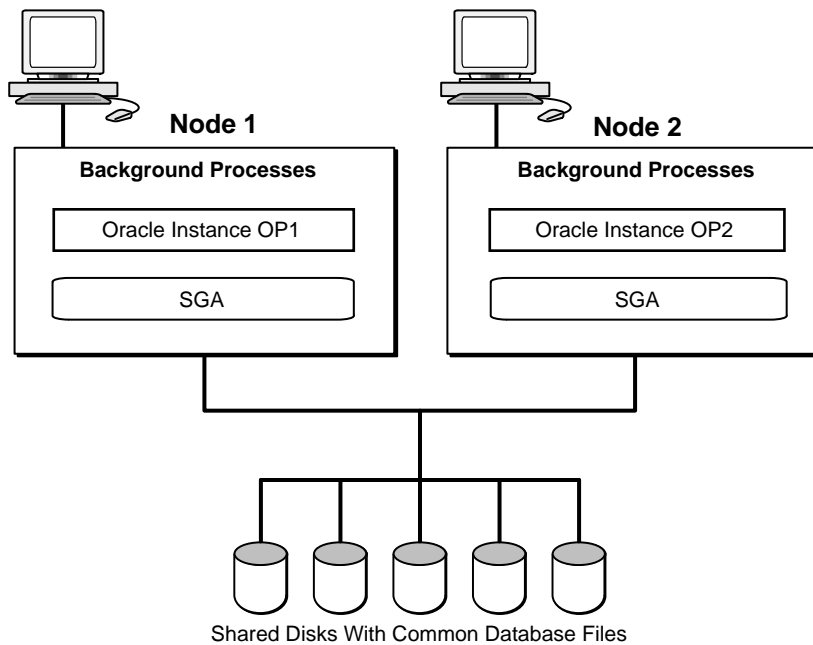
Data Files

A database is logically divided into tablespaces that contain all data stored in the database. Tablespaces, in turn, are made up of one or more data files.

With Oracle Parallel Server, all participating instances access the same data files.

[Figure 1-2](#) shows the relationship between two Oracle instances and the shared disks on which the data files are stored:

Figure 1-2 Oracle Parallel Server Instance Architecture



Oracle Parallel Server Features

Some of Oracle Parallel Server features to take note of include:

Feature	Description
Integrated Distributed Lock Manager (IDLM)	Maintains a list of system resources and provides locking mechanisms to control allocation and modification of Oracle resources. Every process interested in the database resource protected by the IDLM must open a lock on the resource. See "Integrated Distributed Lock Manager" on page 1-15.
Parallel Cache Management (PCM)	Provides instance locks (with minimal use of the IDLM) that cover one or more data blocks of any class: data block, index blocks, undo blocks, segment headers, and so on. Oracle Parallel Server uses these instance locks to coordinate access to shared resources. The IDLM maintains the status of the instance locks See "Parallel Cache Management" on page 1-17.
Oracle Parallel Query	Ships queries between nodes so that multiple nodes can execute on behalf of a single query. See "Oracle Parallel Query" on page 1-20.

See *Oracle8i Parallel Server Concepts and Administration* for further information about these and other features.

Integrated Distributed Lock Manager

The Integrated Distributed Lock Manager (IDLM) maintains a list of system resources and provides locking mechanisms to control allocation and modification of Oracle *resources*. Resources are structures of data. The IDLM does not control access to tables or anything in the database itself. Every process interested in the database resource protected by the IDLM must open a lock on the resource.

Oracle Parallel Server uses the IDLM facility to coordinate concurrent access to resources, such as data blocks and rollback segments, across multiple instances. The Integrated Distributed Lock Manager facility has replaced the external Distributed Lock Manager which was used in earlier releases of Oracle Server.

The IDLM uses the LMON and LMDN processes. LMON manages instance and processes deaths and associated recovery for the IDLM. In particular, LMON handles the part of recovery associated with global locks. The LMDN process handles remote lock requests (those which originate from another instance).

The IDLM:

- Provides a distributed architecture. In the distributed architecture each node in the cluster (or each instance of an Oracle database) participates in global lock management and manages a piece of the global lock database. The lock database is distributed among all the participants.
- Provides continual service and maintains the integrity of the lock database in the event of multiple node and instance failures.
- Maintains information about the locks on all nodes that are interested in a given resource. In this situation, the IDLM usually nominates one node to manage all relevant information about the resource and its locks.
- Performs deadlock detection.
- Synchronizes modifications made to the Oracle data files so that no changes are lost.
- Synchronizes disk block access between multiple nodes.

The IDLM is a resource manager and, thus, does *not* control access to the database.

Example A node in a cluster needs to modify block number *n* in the database file. At the same time, another node needs to update the same block *n* to complete a transaction.

Without the IDLM, *both* nodes update the same block at the same time. With the IDLM, only one node is allowed to update the block. The other node must wait. The IDLM ensures that only one instance has the right to update a block at any one time. This provides data integrity by ensuring that all changes made are saved in a consistent manner.

Interaction with CM

The IDLM uses CM to determine which instances are active. When the instance is started, the LMON and LMDN processes are started and the IDLM registers with CM. The IDLM deregisters with CM when the database is shutdown.

Parallel Cache Management

Parallel Cache Management (PCM) provides instance locks (with minimal use of the IDLM) that cover one or more data blocks of any class: data block, index blocks, undo blocks, segment headers, and so on. Oracle Parallel Server uses these instance locks to coordinate access to shared resources. The IDLM maintains the status of the instance locks.

PCM locks ensure cache coherency by forcing instances to acquire a lock before modifying or reading any database block. PCM locks allow only one instance at a time to modify a block. If a block is modified by an instance, the block must first be written to disk before another instance can acquire the PCM lock, read the block, and modify it.

Example If node 1 needs access to data that is currently in node 2's buffer cache, node 1 can submit a request to the IDLM. Node 2 then writes the needed blocks to disk. Only then is Node 1 notified by the IDLM to read updated and consistent data from the disk.

PCM Lock Implementation

You use the initialization parameter `GC_FILES_TO_LOCKS` to specify the number of PCM locks which cover the data blocks in a data file or set of data files. The smallest granularity is one PCM lock per data block; this is the default. PCM locks usually account for the greatest proportion of instance locks in a parallel server.

PCM locks are implemented in two ways:

PCM Lock	Description
hashed locks	<p>Hashed locks are pre-allocated and statically assigned to one or more blocks at startup time. Hashed locks offer the advantage of no overhead for lock allocation when a data block is accessed.</p> <p>The first instance which starts up creates an IDLM resource and an IDLM lock (in null mode) on the IDLM resource for each hashed PCM lock. The first instance initializes each lock. The instance then proceeds to convert IDLM locks to other modes as required. When a second instance requires a particular IDLM lock, it waits until the lock is available and then converts the lock to the mode required. The total number of locks that can be allocated is limited by system resources. This usually means that multiple blocks have to be covered by the same lock. In other words, there is a low lock granularity. This might result in false ping-pong. The startup of the instance also requires more time, since all the lock resources have to be allocated at startup time.</p> <p>Typically, hashed locks are never released; each will stay in the mode in which it was last requested. You can, however, specify releasable hashed locks by using the R option with the GC_FILES_TO_LOCKS parameter. Releasable hashed PCM locks are taken from the pool of GC_RELEASABLE_LOCKS.</p>
fine-grain locks	<p>This is the default implementation.</p> <p>With fine-grain locking, locks are dynamically allocated at block-access time. The resources for the lock are only allocated during the time the lock is needed and are released when the lock is released. This makes it possible to achieve very high-lock granularity. If resource minimization is the goal, fine-grain locks can also cover multiple blocks, but are still allocated dynamically.</p> <p>Since locks are allocated only as required, the instance can start up much faster than with hashed locks. An IDLM resource is created and an IDLM lock is obtained only when a user actually requests a block. Once a fine-grain lock has been created, it can be converted to various modes as required by various instances.</p> <p>Typically, fine-grain locks are releasable: An instance can give up all references to the resource name during normal operation. You can, however, allocate fixed locks in a fine grained manner with the GC_FILES_TO_LOCKS parameter. Creating a 1-to-1 ratio of locks-to-blocks creates DBA locking.</p>

See the *Oracle8i Parallel Server Concepts and Administration* for further information about PCM locks.

It is possible to have both fine-grain and hashed locking enabled at the same time.

Below is a comparison of both PCM locks:

Hash PCM Locks	Fine-Grain PCM Locks
<ul style="list-style-type: none"> ■ Allocated at instance startup, resulting in a slower startup. ■ Released only at instance shutdown. ■ Statically hashed to blocks at startup time, requiring more memory. 	<ul style="list-style-type: none"> ■ Allocated when user requests a block, resulting in faster instance startup. ■ Dynamically re-used by other blocks, requiring less memory.

Choosing a Lock

Use the table below to choose a PCM lock:

When to use hashed locks...	When to use fine-grain locks...
<ul style="list-style-type: none"> ■ Data is mostly read-only. ■ Data can be partitioned. ■ Large set of data is modified by a relatively small set of instances. 	<ul style="list-style-type: none"> ■ Small amount of data is updated by many instances. ■ There is not enough memory for the configuration of hash locking.

Oracle Parallel Query

With Oracle Parallel Query, Oracle can divide the work of processing certain types of SQL statements among multiple query server processes.

Architectural Overview

When parallel execution is not being used, a single server thread performs all necessary processing for the sequential execution of a SQL statement. For example, to perform a full table scan (such as `SELECT * FROM EMP`), one thread performs the entire operation.

Oracle Parallel Query performs the operations in parallel using multiple parallel processes. One process, known as the *parallel coordinator*, dispatches the execution of a statement to several *parallel server processes* and coordinates the results from all the server processes to send the results back to the user.

The parallel coordinator breaks down execution functions into parallel pieces and then integrates the partial results produced by the parallel server processes. The number of parallel server processes assigned to a single operation is the degree of parallelism for an operation. Multiple operations within the same SQL statement all have the same degree of parallelism.

Oracle Parallel Query on Oracle Parallel Server

Oracle Parallel Server provides the framework for the Parallel Query Option to work between nodes. The Oracle Parallel Query behaves the same way in Oracle with or without the Parallel Server Option. The only difference is that Oracle Parallel Server enables Oracle Parallel Query to ship queries between nodes so that multiple nodes can execute on behalf of a single query. Here, the server breaks the query up into smaller operations that run against a common database which resides on shared disks. Because it is performed by the server, this parallelism can occur at a low level of server operation, rather than at an external SQL level.

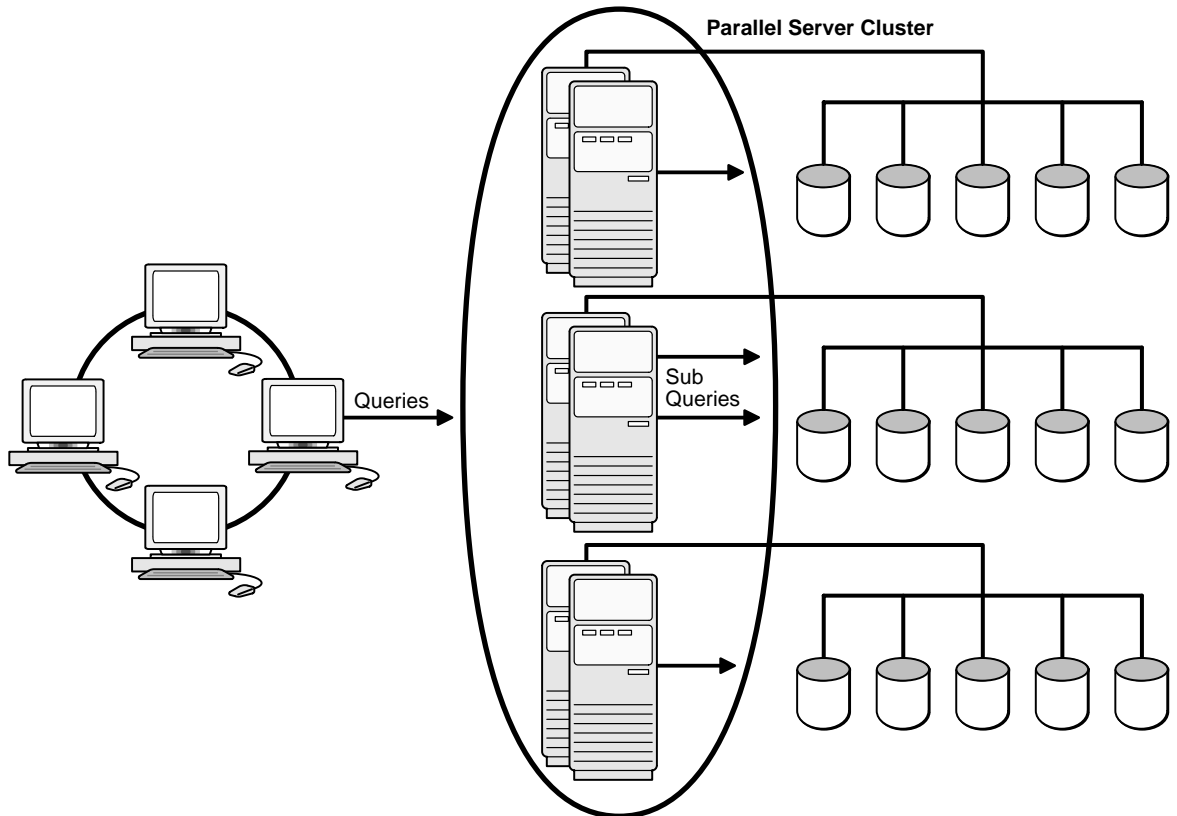
In some applications, an individual query often consumes a great deal of CPU resource and disk I/O (unlike most online insert or update transactions). To take advantage of multi-processing systems, the data server must parallelize individual queries into units of work which can be processed simultaneously.

If the query were not processed in parallel, disks would be read serially with a single I/O. A single CPU would have to scan all rows in a table. With the query parallelized, disks are read in parallel, with multiple I/Os.

Several CPUs can each scan a part of the table in parallel, and aggregate the results. Parallel query benefits not only from multiple CPUs but also from greater I/O bandwidth availability.

Oracle Parallel Query can run with or without the Oracle Parallel Server. Without the Oracle Parallel Server option, Oracle Parallel Query cannot perform multi-node parallelism. Oracle Parallel Server optimizes Oracle8i Enterprise Edition running on clustered hardware, using a parallel cache architecture to avoid shared memory bottlenecks in OLTP and decision support applications.

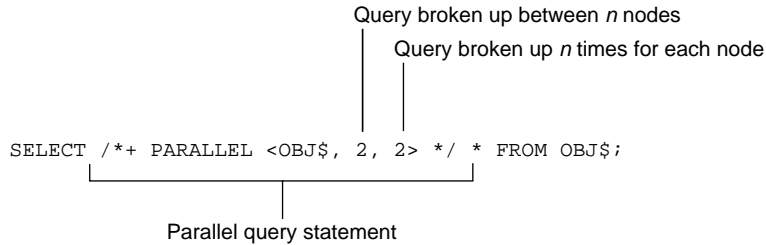
Figure 1-3 Parallel Query Operation



Oracle Parallel Query Statement

Oracle Parallel Query within Oracle Parallel Server performs parallelism within a node and among nodes via the parallel query slave processes on each node.

A sample SQL statement is shown below:



After you have run a query, you can use the information derived from `V$PQ_SYSSTAT` to view the number of slave processes used, and other information for the system.

See the *Oracle8i Concepts* for further information about parallel execution.

Pre-Installation

This chapter describes pre-installation requirements for Oracle Parallel Server.

Specific topics discussed are:

- [System Installation Requirements](#)
- [Setting Up Raw Devices](#)
- [Pre-Installation Steps](#)

System Installation Requirements

Verify that your system meets the installation requirements described in the following sections before you install.

- [Hardware and Software Requirements for Oracle8i Parallel Server Node](#)
- [Hardware and Software Requirement for Oracle Enterprise Manager](#)
- [Shared Disk Subsystem](#)

Hardware and Software Requirements for Oracle8i Parallel Server Node

Very hardware software requirements for each node:

Hardware

Each node in a cluster requires the following hardware:

- Operating system specific hardware, as described in platform-specific installation guides.
- external shared hard disk

Software

Each node in a cluster requires the following software:

- Operating-system specific software, as described in platform-specific installation guides.
- Operating System Dependent layer from a vendor that has passed certification
- Oracle8i Enterprise Edition
- Net8 Server
- Oracle Parallel Server Option
- Oracle Intelligent Agent release 8.1.5 if using Oracle Enterprise Manager

The Oracle Intelligent Agent may be installed from the Oracle8i Enterprise Edition CD-ROM

- Oracle Data Gatherer release 8.1.5 if using the Oracle Diagnostics Pack's Oracle Performance Manager product to generate performance reports

The Oracle Data Gatherer is automatically installed with the Oracle Intelligent Agent.

- ❑ One of the following Web browsers to view online documentation:
 - Netscape Navigator Version 3.0 or later.
<http://www.netscape.com>
 - Microsoft Internet Explorer Version 3.0 or later.
<http://www.microsoft.com>

Hardware and Software Requirement for Oracle Enterprise Manager

Oracle Enterprise Manager version 2 is a management framework consisting of a Console, a suite of tools and services, and a network of management servers and Oracle Intelligent Agents.

You can run the individual Oracle Enterprise Manager components on separate machines or combine different components on separate machines to collaboratively manage the complete Oracle environment.

The components are listed below:

- [Console and Database Administration Applications](#)
- [Management Server](#)
- [Repository database](#)
- [Managed targets with agents throughout the network](#)

Console and Database Administration Applications

The Console works with the Oracle Intelligent Agent (installed on the nodes) to perform database administration from a single console.

If both the Console and Database Administration Applications are on their own machines, Net8 installation and configuration is required to log on to the repository or managed nodes. Net8 is not required on nodes running only the Console.

Management Server

The Management Server performs functions requested by the Console. Each Management Server manages exactly one domain. The domain is the logical grouping of machines for administrative purposes.

If the Management Server is on its own machine separate from the repository database, Net8 installation and configuration is required to communicate with the repository database.

Repository database	<p>A repository is a set of tables in an Oracle database which stores data required by Oracle Enterprise Manager.</p> <p>The repository must be stored in an Oracle database version 7.3.3 or above.</p>
Managed targets with agents throughout the network	<p>A managed target is any service or entity which can be centrally managed by Oracle Enterprise Manager. Examples are nodes, databases, web servers, listeners, and paging services.</p> <p>Each Oracle Parallel Server must have a release 8.1.5 or earlier Oracle Intelligent Agent installed.</p>

Operating System

- Windows NT version 4.0
- Solaris 2.6

Note: The Oracle Enterprise Manager Console and DBA Applications may also run on Windows 95 or Windows 98. The OMS and repository cannot run on these platforms.

Enterprise Manager Software

- Oracle Enterprise Manager version 2
- Oracle Diagnostics Pack version 2

The Oracle Diagnostics Pack includes the Oracle Performance Manager application, which allows you to choose from a variety of graphic performance statistical charts for parallel servers.

Note: You may also use prior releases of Oracle Enterprise Manager and Oracle Diagnostics Pack for a release 8.1.5 database. See Oracle Enterprise Manager documentation for compatibility information.

- Net8 Client on the machine where the Diagnostics Pack is installed. Net8 Client is also required if the OMS and Database Administration Applications and Console are on their own machines.

Supported Oracle Database Versions as Repositories

8.1.5, 8.0.5, 8.0.4, 8.0.3, 7.3.4, 7.3.3 on database repository machine

Supported Oracle Intelligent Agents

8.1.5 on all Oracle Parallel Server nodes

The Oracle Intelligent Agent may be installed from the Oracle8i Enterprise Edition CD-ROM.

See *Oracle Enterprise Manager Installation* for detailed disk space and RAM requirements.

Shared Disk Subsystem

Oracle Parallel Server requires a shared disk subsystem to contain shared partitions that are raw. All Oracle8i data, log, and control files are placed on shared raw partitions.

Note: Each instance of an Oracle Parallel Server database has its own log files, but control files and data files are shared by instances in the cluster. However, log files must be accessible/readable by other instances.

Setting Up Raw Devices

Data files and control files must be shared by all instances. Each instance has its own log files, but all instances must have access to all log files during recovery. UNIX and Windows NT clusters do not provide access to a shared file system between all nodes of a cluster. As a result, all files associated with a database must be built on raw devices.

The database is created as a part of the installation process. Because of this, the raw devices must be set up prior to installation in order for the data files, control files and all logs to be copied to the raw devices.

A precise number of raw devices is required for the Oracle Database Configuration Assistant to succeed in creating the database:

- six for data files
- one for the Oracle8i *interMedia* (if used)
- two for control files
- two for redo log files for each node

The table below lists the raw devices that must be created and the minimum size that the Oracle Database Configuration Assistant expects. When creating a raw device, ensure the size is a little larger than indicated below to account for future growth.

Raw Device Must be Created For...	Size to Use on UNIX	Size to Use on Windows NT
DR tablespace for Oracle8i <i>interMedia</i> option	80 MB	101 MB
SYSTEM tablespace for system data file	120 MB	101 MB
TEMP tablespace for temporary data file	10 MB	101 MB
RBS tablespace for rollback data file	15 MB	50 MB
INDX tablespace for indexes	10 MB	50 MB
OEM_REPOSITORY tablespace for the Oracle Enterprise Manager repository	5 MB	50 MB
USERS tablespace for user workspace	10 MB	20 MB

Raw Device Must be Created For...	Size to Use on UNIX	Size to Use on Windows NT
first control file	7 K	2 MB
second control file	7 K	2 MB
two redo log files for each node	6 K	3 MB

The creation of raw devices is platform specific:

- [UNIX](#)
- [Windows NT](#)

UNIX

On UNIX platforms, creation of raw devices is as follows:

1. Create raw devices, as explained in the *Oracle8i Administrator's Reference* for Sun Solaris, HP 9000 or AIX-based systems.

Note: Raw devices must be created by the `root` user on UNIX platforms.

For each raw device, use a file name that makes sense for the data file it will store, for example:

Raw Device	Example File Name
Oracle8i <i>interMedia</i> option raw device	dr01.dbf
SYSTEM tablespace raw device	system01.dbf
TEMP tablespace raw device	tmp01.dbf
RBS tablespace raw device	rbs01.dbf
INDX tablespace raw device	indx01.dbf
OEM_REPOSITORY tablespace raw device	oemrep01.dbf
USERS tablespace raw device	user01.dbf
first control file raw device	control01.clt

Raw Device	Example File Name
second control file raw device	control02.clt
redo log files for each node	<i>redothread_id_number.log</i> <i>thread_id</i> is the thread ID of the node and <i>number</i> is the log number (1 or 2) for the node

2. On the node you intend to later run the Oracle Universal Installer, create an ASCII file with entries for each raw device file name, as explained in ["Identifying Raw Devices"](#) on page 2-10.
3. On the same node with the ASCII file, create an environment variable named DBCA_RAW_CONFIG that points to the location of this ASCII file. When the Oracle Database Configuration Assistant creates the database, it looks for the environment variable, reads in the ASCII file, and stores the files in the raw devices.

Windows NT

On Windows NT, creation of raw devices is as follows:

1. Create raw devices with the Disk Administrator, as explained in the *Oracle8i Parallel Server Getting Started* for Windows NT.

Note: When creating raw partitions on Windows NT, you must unassign the drive letters assigned to the raw partitions. This will free up drive letters for other purposes, such as mapping network drives.

2. On each node, create an ASCII named *DB_NAME.TBL* with symbolic link names associated with each raw device, as explained in "[Identifying Raw Devices](#)" on page 2-10.
3. Run the Oracle SETLINKS utility against the ASCII file to create the symbolic link names. When the Oracle Database Configuration Assistant creates the database, it verifies the symbolic links names have been created, and stores the files in the raw devices.

Identifying Raw Devices

Once raw devices are set up, the Oracle Database Configuration must have a way to map file to them when the database is create. This identification is platform-specific and must be set up prior to installation:

- [UNIX](#)
- [Windows NT](#)

UNIX

UNIX platforms identify raw devices through files names in an ASCII file that contains entries that follow the format of:

database_object raw_device_file

Separate entries should exist for the following database objects. The file names are names you created in step 1 in [UNIX](#) on page 2-7

Database Object	Description
dr	DR tablespace for Oracle8i <i>interMedia</i> option
sys1	SYSTEM tablespace for system data file
tmp1	TEMP tablespace for temporary data file
rbs1	RBS tablespace for rollback data file
indx1	INDX tablespace for indexes
oemrep1	OEM_REPOSITORY tablespace for the Oracle Enterprise Manager repository
usr1	USERS tablespace for user workspace
ctl1	first control file
ctl2	second control file
<i>logthread_id_number</i>	redo log file
<i>thread_id</i> is the thread ID of the node and <i>number</i> is the log number (1 or 2) for the node	Two entries for each node required. Entries for the first node would like: log1_1 log1_2 Entries for the second node would like: log2_1 log2_2

Example ASCII File The ASCII file should look like the example below for a two-node cluster:

```
dr          device/path/dr01.dbf
sys1       device/path/system01.dbf
tmp1       device/path/tmp01.dbf
rbs1       device/path/rbs01.dbf
indx       device/path/indx01.dbf
oemrep1    device/path/oemrep01.dbf
usr1       device/path/user01.dbf
ctl1       device/path/control01.clt
ctl2       device/path/control02.clt
log1_1     device/path/redo1_1.log
log1_2     device/path/redo1_2.log
log2_1     device/path/redo2_1.log
log2_2     device/path/redo2_2.log
```

Windows NT

Windows NT uses symbolic link names to identify their raw devices in an ASCII file named *DB_NAME.TBL* that follow the format:

```
db_name_link_name      \Device\HarddiskX\PartitionX
```

Symbolic link names on Windows NT must be an exact name for the following:

- [Data Files](#)
- [Control Files](#)
- [Redo Log Files](#)

Data Files When creating the raw partitions for the data files, you must use the following symbolic link names:

Symbolic Link Names	Tablespace	Optional?
<i>db_name_dr</i>	DR tablespace for Oracle8i <i>interMedia</i> option	Yes
<i>db_name_sys1</i>	SYSTEM tablespace for system data file	No
<i>db_name_tmp1</i>	TEMP tablespace for temporary data file	No
<i>db_name_rbs1</i>	RBS tablespace for rollback data file	No

Symbolic Link Names	Tablespace	Optional?
<i>db_name_indx1</i>	INDX tablespace for indexes	No
<i>db_name_oemrep1</i>	OEM_REPOSITORY tablespace for the Oracle Enterprise Manager repository	No
<i>db_name_usr1</i>	USERS tablespace for user workspace	No

Control Files When creating the raw partitions for the control files, you must use the following symbolic link names:

- *db_name_ct11*
- *db_name_ct12*

Redo Log Files Two redo log files are required for each node. For a two-node cluster, these file symbolic link names must be:

db_name_logthread_id_number, where *thread_id* is the thread ID of the node and *number* is the log number (1 or 2) for the node.

Each node's log number starts at 1:

- The first node selected will be *db_namelog1_1* and *db_namelog1_2*
- The second node selected will be *db_namelog2_1* and *db_namelog2_2*

Example ASCII File On Windows NT, the ASCII file should look like the example below for a two-node cluster. OP is the name of the database and X represents hard disk or partition numbers. The name of the database must be known prior to installation to create this file.

```
OP_dr           \Device\HarddiskX\PartitionX
OP_sys1        \Device\HarddiskX\PartitionX
OP_tmp1        \Device\HarddiskX\PartitionX
OP_rbs1        \Device\HarddiskX\PartitionX
OP_indx1       \Device\HarddiskX\PartitionX
OP_oemrep1     \Device\HarddiskX\PartitionX
OP_usr1        \Device\HarddiskX\PartitionX
OP_ct11        \Device\HarddiskX\PartitionX
OP_ct12        \Device\HarddiskX\PartitionX
OP_log1_1      \Device\HarddiskX\PartitionX
OP_log1_2      \Device\HarddiskX\PartitionX
OP_log2_1      \Device\HarddiskX\PartitionX
OP_log2_2      \Device\HarddiskX\PartitionX
```

Pre-Installation Steps

Perform the following steps to prior to installation:

1. Install a vendor-supplied Operating System Dependent layer that passed certification. See your vendor supplied Operating System Dependent la
2. Perform diagnostics on the clusterware as described in your vendor documentation.
3. Create raw devices:

On UNIX, perform these tasks:

1. Create raw devices, following instructions in the *Oracle8i Administrator's Reference*.
2. On the node from which you intend to run the Oracle Universal Installer, create an ASCII file with entries for each data file, control file and redo log file. For an example file, see "[Example ASCII File](#)" on page 2-11.
3. Create an environment variable named `DBCA_RAW_CONFIG` that points to the ASCII file .

On Windows NT, perform these tasks:

1. Create raw devices, following instructions in the *Oracle8i Parallel Server Getting Started* for Windows NT.
2. On each node, insert the Oracle8i Enterprise Edition CD-ROM.
3. On each node, copy the SETLINKS executable and the TEMPLATE.TBL file you will need from the `\ops_preinstall` directory to a temporary directory.
4. On each node, change TEMPLATE.TBL to the `DB_NAME.TBL`, where `DB_NAME` is the database name you intend to use.
5. On each node, change `<db_name>` entries to the database name you intend to use, and edit the partition and hard disk numbers appropriately in the *.TBL file. For an example file, see "[Example ASCII File](#)" on page 2-12.
6. On each node, un the *.TBL file through the SETLINKS utility:

```
setlinks /f:dbname.tbl
```

See the *Oracle8i Parallel Server Getting Started* for Windows NT for further information on these steps.

4. For UNIX clusters, perform the following as the `root` user:
 - a. Make sure you have an OSDBA group defined in the `/etc/group` file on all nodes of the cluster. The OSDBA group name and number (and OSOPER group if you plan to designate one during installation) must be identical for all nodes of a UNIX cluster accessing a single database. The default UNIX group name for the OSDBA and OSOPER groups is `dba`.
 - b. Create an *oracle* account on each node of the cluster so that:
 - the account is a member of the OSDBA group
 - the account is used only to install and update Oracle software
 - the account has write permissions on remote directories
 - c. Create a mount point directory on each node to serve as the top of your Oracle software directory structure so that:
 - the name of the mount point on each node is identical to that on the initial node
 - the *oracle* account has read, write, and execute privileges
 - d. On the node from which you plan to run the Oracle Universal Installer, set up user equivalence by adding entries for all nodes in the cluster to the `.RHOSTS` file of the *oracle* account, or the `/etc/hosts.equiv` file.
- Exit the `root` account when you are done.

5. For UNIX clusters, as the *oracle* account, check for user equivalence for the oracle account by performing a remote login (`rlogin`) to each node in the cluster. If you are prompted for a password, the *oracle* account has not been given the same attributes on all nodes. The Oracle Universal Installer cannot use the `rcp` command to copy Oracle products to the remote directories without user equivalence.

Note: UNIX clusters also require environment setup similar to a single-instance environment. For these instructions and other platform-specific Oracle Parallel Server pre-installation instructions, see the *Oracle8i Installation Guide* for Sun Solaris, HP 9000 or AIX-based systems.

Installation

This chapter describes using the Oracle Universal Installer to install a new Oracle8i Enterprise Edition and the Oracle Parallel Server Option installation.

Specific topics discussed are:

- [Installation Types](#)
- [Installation Procedure](#)
- [Migrating or Upgrading to Release 8.1](#)
- [Multiple Oracle Homes](#)

Installation Types

The type of installation and the type of database you choose to install determines the how you proceed with installation and configuration decisions.

When you run Oracle Universal Installer, you can choose to install Oracle8i Enterprise Edition and the Oracle Parallel Server Option with the following installation types:

Installation Type	Description
Typical	<p>Installs a preconfigured starter database, licensable Oracle options and cartridges (including Oracle Parallel Server), networking services, Oracle8i utilities, and online documentation. This type of installation is recommended for new users or experienced users that want the complete database package.</p> <p>At the end of installation, the Oracle Database Configuration Assistant runs to create and start Oracle instances, copy data files, and initialize the database.</p>
Custom	<p>Note: Custom is only supported on Windows NT in release 8.1.5.</p> <p>Allows you to selectively install any product from the CD-ROM, including Oracle8i Enterprise Edition and Oracle Parallel Server on any subset of nodes in the cluster. If Oracle8 Enterprise Edition is chosen, you can choose to start the Oracle Database Configuration Assistant. From the Oracle Database Configuration Assistant, you may create a:</p> <ul style="list-style-type: none">■ starter database■ Online Transaction Processing (OLTP) database■ Decision Support System (DSS) database■ Hybrid database <p>If you choose not to run the Oracle Database Configuration Assistant, you may run it at a later time or manually configure Oracle Parallel Server.</p> <p>If you choose not to install a database, you can install the Oracle Parallel Server Option and manually create the database.</p>

Note: The Minimal install type does not support the Oracle Parallel Server Option.

Installation Procedure

During installation, software components are installed on the node from which the Oracle Universal Installer is run, and pushed to the other selected nodes in the cluster.

To install Oracle8i Enterprise Edition and the Oracle Parallel Server Option:

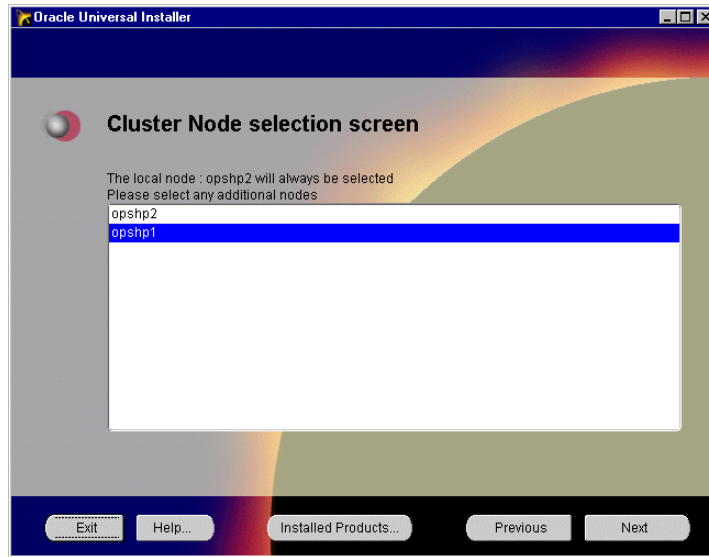
1. Ensure the tasks in "[Pre-Installation Steps](#)" on page 2-13 were performed.
2. From the node where you intend to run the Oracle Universal Installer, verify you have administrative privileges to the other nodes.

Operating System	Verification Method
On UNIX	Perform a remote copy (<code>rCP</code>) to the other nodes.
On Windows NT	<p>Enter the following command for each node that is a part of the cluster:</p> <pre>NET USE \\host_name\C\$</pre> <p>where <i>host_name</i> is the host name defined in the DefinedNodes registry value for Cluster Manager.</p> <p>A successful connection results in "<i>The command completed successfully.</i>"</p> <p>Oracle Corporation recommends using the same user name and password on each node in a cluster or use a domain user name. If you use a domain user name, log on under a domain with username and password which has administrative privileges on each node.</p>

3. Install Oracle8i Enterprise Edition on *one* node in the cluster, following instructions in the:
 - *Oracle8i Installation Guide* for Sun Solaris, HP 9000 or AIX-based systems
 - *Oracle8i Enterprise Edition Installation for Windows NT*
4. Take note of the following tasks to perform for a successful Oracle Parallel Server installation:
 - a. Select Oracle8i Enterprise Edition from the *Available Products* page.
 - b. Select either Typical or Custom from the *Installation Types* page. See "[Typical Installation](#)" on page 3-5 or "[Custom Installation](#)" on page 3-6.
 - c. If performing a Custom installation, ensure the Oracle Parallel Server option under the Oracle Database Options folder is selected in the *Available Product Components* page:

The Oracle Universal Installer will not make Oracle Parallel Server Option visible unless Cluster Manager clusterware was properly configured. See your OSD vendor documentation for further information.

- d. Select the nodes to which you want the software installed from the *Cluster Node selection screen* page:



The node on which you are running the installation is selected whether or not you select it.

If the list of nodes does not include nodes you expect, the vendor-supplied clusterware is either not installed, running, or configured properly. See your vendor documentation for further information.

If clusterware is not installed, click Previous in the Oracle Universal Installer, install Cluster Manager on those nodes, then click Next in the Oracle Universal Installer. The nodes should now be listed.

- e. In the *Database Identification* page, enter an appropriate global database name, database name and domain, for the Oracle Parallel Server, such as *OP.domain*, and accept or change the prefix for the Oracle System Identifier (SID) that will be used in the SID field.

Each instance will have a SID that consists of the common prefix entered here and a thread ID that is automatically generated. For example, if OP is

entered, the first instance in the cluster will have a SID of OP1, and the second instance will have a SID of OP2.

During installation no message appears to indicate components are installed to the other nodes.

Typical Installation

If you choose a Typical installation type, the Oracle Database Configuration Assistant is run automatically to create the database with no user input. The information entered in the *Database Identification* page is used by the Oracle Database Configuration Assistant to identify the database and each instance's SID.

The Oracle Database Configuration Assistant verifies the raw devices file names were created for each tablespace that will be created. Raw devices and file names should have been configured in "[Setting Up Raw Devices](#)" on page 2-6. If the file names were not set up, the Oracle Database Configuration Assistant will fail to create the database.

Among the files installed, Typical results in the creation of these files:

- `INITDB_NAME.ORA` for general parameters shared by all instances
- `INITSID.ORA` files for each instance in the cluster

See "[Understanding the Initialization Files](#)" on page 6-2 for further information about these files.

Note: If the Oracle Database Creation Assistant fails to create the database, see "[Cleaning Up the Registry After an Oracle Database Configuration Assistant Failure on Windows NT](#)" on page C-2.

Custom Installation

During the a Custom installation, the following products run, prompting for user input:

- [Net8 Configuration Assistant](#)
- [Oracle Database Configuration Assistant](#)

Net8 Configuration Assistant The Net8 Configuration Assistant prompts you to configure the network. The Net8 Configuration Assistant only allows you to configure some aspects of the network for the node the Oracle Universal Installer is running on. Follow these recommendations:

1. Follow the Net8 Configuration Assistant prompts to configure the listener for the node by identifying a listener name and listener protocol address that uses TCP/IP on port 1521. TCP/IP is required for Oracle Enterprise Manager.

Note: If you do not choose the registered port of 1521, you must configure the LOCAL_LISTENER parameter in the INITDB_NAME.ORA file and resolve it with a net service name entry in the TNSNAMES.ORA file. For further information about configuring non-default port numbers in LISTENER.ORA, see Chapter 6, "Configuring Naming Methods and the Listener" in the *Net8 Administrator's Guide*.

2. When asked to configure naming methods, select *No, do not configure naming methods*. Oracle Parallel Server requires more advanced naming method configuration than is available through the Net8 Configuration Assistant.

See "[Configuring Net8 for Nodes](#)" on page 4-12 to complete network configuration for this node and other nodes in the cluster.

Oracle Database Configuration Assistant The information entered in the *Database Identification* page is used by the Oracle Database Configuration Assistant to identify the database name and each instance's SID.

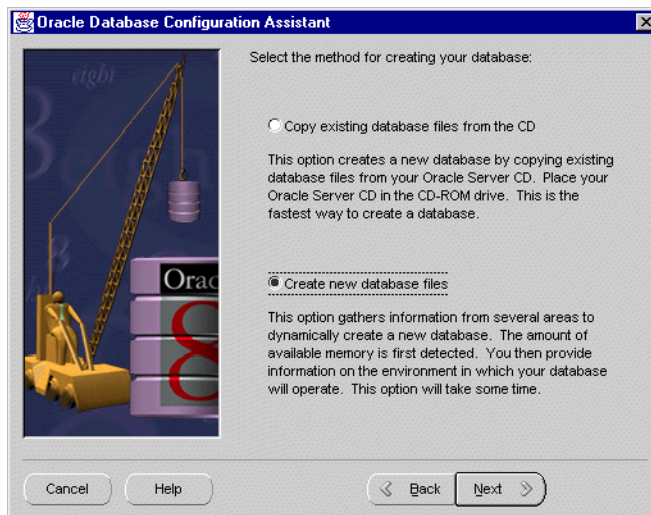
The Oracle Database Configuration Assistant verifies the raw devices file names were created for each tablespace that will be created. Raw devices and file names should have been configured in "[Setting Up Raw Devices](#)" on page 2-6. If the file names were not set up, the Oracle Database Configuration Assistant will fail to create the database.

If you chose a Custom installation type, you are prompted with a choice to create a database:

- using the Oracle Database Configuration Assistant during installation
- sometime after installation using either the Oracle Database Configuration Assistant or other manual methods.

See "[Creating the Database](#)" on page 4-3 to create a database with the Oracle Database Configuration Assistant after installation.

If you choose to create a database using the Oracle Database Configuration Assistant, the following page appears:



Option	Description
Copy existing database files from the CD	Creates a standard starter database with parameters for each instance in <code>INITSID.ORA</code> and common parameters shared from node-to-node in <code>INITDB_NAME.ORA</code> .
Create new database files	Customizes the creation of the following database types. <ul style="list-style-type: none"> ■ Online Transaction Processing (OLTP) database ■ Decision Support System (DSS) database ■ Hybrid database (combination of OLTP and DSS)

Respond to instructions in each Oracle Configuration Database Assistant page, then click Next when you are ready to continue to the next screen. When the last page displays, click Finish to start the creation of the Oracle Parallel Server database.

Migrating or Upgrading to Release 8.1

If the Oracle Universal Installer detects an earlier version of an Oracle database on your hard drive, you will be prompted to migrate or upgrade the database to release 8.1. Do not click the *Migrate an Existing Database* check box, as the Oracle Data Migration Assistant does not support Oracle Parallel Server. Instead, use the Migration utility to migrate after installation, as described in the *Oracle8i Migration* guide.

Note: Migrating or upgrading on Windows NT involves disabling the OraclePGMSService and dealing with the Operating System Dependent layer. The handling of the Operating System Dependent layer can be quite involved, especially during a migration. These instructions are not covered in the *Oracle8i Migration* guide. See *Oracle8i Parallel Server Getting Started* for Windows NT for complete migration and upgrade steps.

Take note of the following:

- Coexistence of different versions of the database is platform-specific. See the appropriate installation guide:
 - *Oracle8i Installation Guide* for Sun Solaris, HP 9000 or AIX-based systems
 - *Oracle8i Enterprise Edition Installation for Windows NT*
- Rolling upgrades—where different databases, or different instances of the same database in Oracle Parallel Server, are upgraded to a new version of Oracle one at a time—are not supported.

Multiple Oracle Homes

Like a regular Oracle8i Enterprise Edition database, the Oracle Parallel Server Option on UNIX supports multiple Oracle homes, a feature that allows you to install one or more releases on the same machine (in multiple Oracle home directories). Windows NT supports multiple Oracle homes, but all the Oracle homes must contain the same release of the Oracle8i Enterprise Edition and the Oracle Parallel Server Option.

Both UNIX and Windows NT require that each node have only one copy of the vendor-supplied Operating System Dependent layer active.

Configuring Oracle Parallel Server

This chapter describes the configuration tasks to create an operating Oracle Parallel Server.

Specific topics discussed are:

- [Configuration Options](#)
- [Creating the Database](#)
- [Starting the Database in Parallel Mode](#)
- [Verifying Instances Are Running](#)
- [Configuring Net8 for Nodes](#)
- [Configuring Net8 for Clients](#)
- [Configuring Net8 with Failover](#)
- [Testing the Net8 Configuration](#)

Configuration Options

The type of installation you chose determines the type of configuration to perform:

If you chose this installation type...	Configuration consists of...
Typical	This installation type requires no configuration on your part, as the Oracle Database Configuration Assistant is run as a part of installation for you.
Custom	<p>If you chose to install a database, you are prompted to run the Oracle Database Configuration Assistant to create a database. It is advisable to use the Oracle Database Configuration Assistant to create a Starter, OLTP, DSS, or Hybrid database. If you choose not to run the Oracle Database Configuration Assistant, you may run it at a later time or manually configure Oracle Parallel Server.</p> <p>If you choose not to install a database, you must manually create the database.</p>

Creating the Database

If you chose not to create a database during a Custom install, you can use the Oracle Database Configuration Assistant after installation to create an Oracle Parallel Server database.

[Step 1: Create and Identify Raw Partitions](#)

[Step 2: Create the Database](#)

Step 1: Create and Identify Raw Partitions

The Oracle Database Configuration Assistant assumes the raw devices were created and names. It will verify raw devices exist prior to database creation. If the raw devices were not set up, the Oracle Database Configuration Assistant will fail to create the database.

If the raw devices have not been created, perform these tasks:

1. Create the raw devices and name them properly, as described in "[Setting Up Raw Devices](#)" on page 2-6.
2. Follow these platform-specific tips:

On UNIX:	On Windows NT:
<ol style="list-style-type: none"> 1. Create an ASCII file with entries for each data file, control file and redo log file. For an example file, see "Example ASCII File" on page 2-11. 2. Create an environment variable named <code>DBCA_RAW_CONFIG</code> that points to the ASCII file on the node from which you intend to run the Oracle Universal Installer. 	<p>On each node:</p> <ol style="list-style-type: none"> 1. Go to the <code>ORACLE_HOME\ops\dbca</code> directory. 2. Change <code>TEMPLATE.TBL</code> to <code>DB_NAME.TBL</code>, where <code>DB_NAME</code> is the database name you intend to use. If you intend to name your database <code>OP</code>, you can use <code>OPT.TBL</code> and avoid having to change the <code><db_name></code> entries in step 2. 3. Change <code><db_name></code> entries to the database name you intend to use, and edit the partition and hard disk numbers appropriately in the <code>*.TBL</code> file. For an example file, see "Example ASCII File" on page 2-12. 4. Run the <code>*.TBL</code> file through the <code>SETLINKS</code> utility: <pre>setlinks /f:dbname.tbl</pre>

See the *Oracle8i Parallel Server Getting Started* for Windows NT for further information about these steps.

Step 2: Create the Database

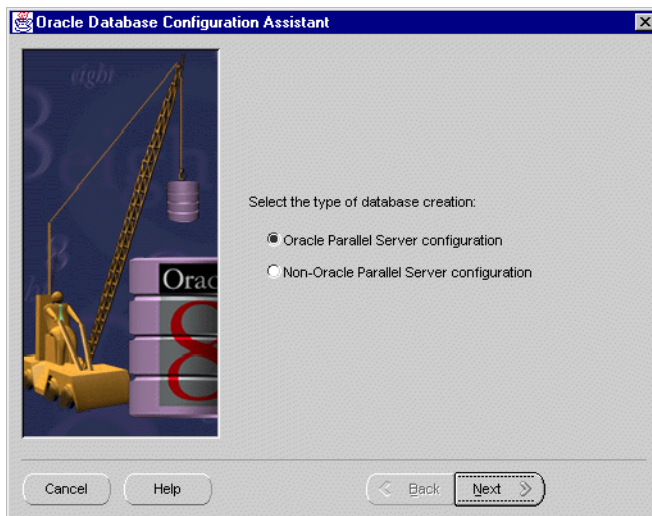
To create a database using the Oracle Database Configuration Assistant:

1. Start the Oracle Database Configuration Assistant on the node from which the Oracle Universal Installer was run:

-On Windows NT, choose Start > Programs > Oracle for Windows NT - [HOME_NAME] > Oracle Database Configuration Assistant.

-On UNIX, run `dbassist` from `$ORACLE_HOME/bin`.

The Oracle Database Configuration Assistant welcome page appears:



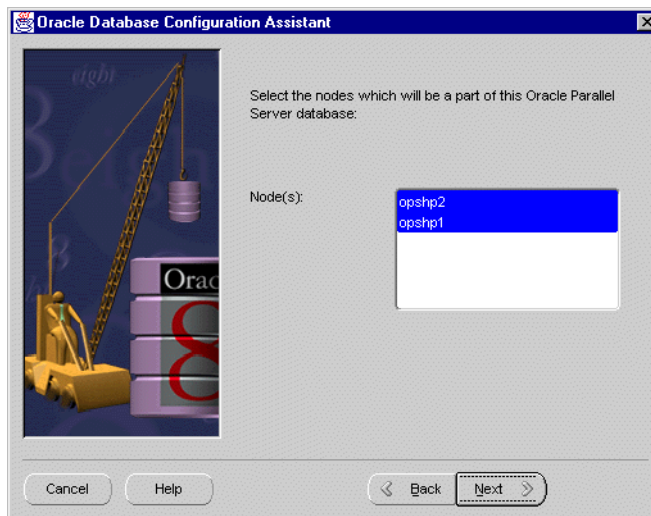
Note: If this page does not display, it means the Oracle Database Configuration Assistant was unable to:

- detect the Lock Manager software or the lists of nodes in the cluster on UNIX platforms
- load the Cluster Manager software on Windows NT

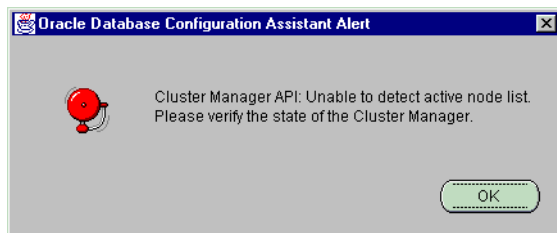
See your Operating System Dependent vendor documentation to resolve the problem, then start the Oracle Database Configuration Assistant again.

2. Select Oracle Parallel Server Configuration, then click Next.

The following page appears:



The error message below may appear if there is a problem with the clusterware:



On UNIX, this message indicates that the Oracle Database Configuration Assistant is unable to detect the lists of nodes in the cluster. See your Operating System Dependent vendor documentation for further information.

On Windows NT, this messages indicates that Cluster Manager (CM) is installed but:

- there is an error calling the API
- the local node is not recognized
- you do not have administrative privilege to any of the nodes

To resolve this error message:

- a. Check to see if CM was installed and configured properly. See your Operating System Dependent layer vendor documentation for further information.
- b. Check if you have administrative privileges on nodes by entering:

```
NET USE \\host_name\C$
```

where *host_name* is the host name defined in the DefinedNodes registry value for CM.

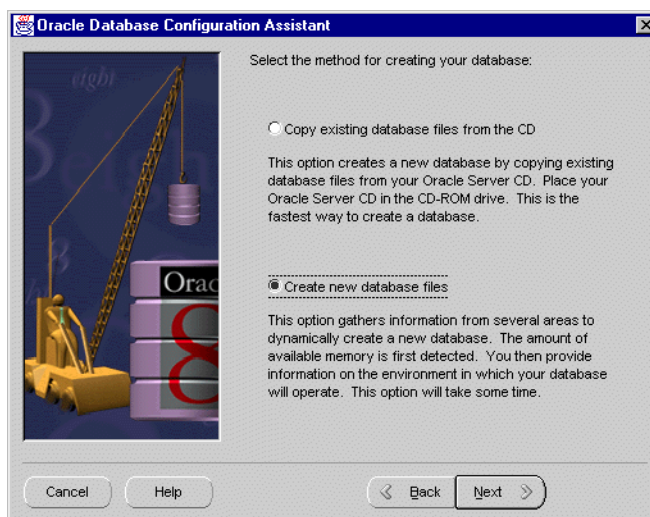
A successful connection results in "*The command completed successfully.*"

Oracle Corporation recommends using the same user name and password on each node in a cluster or use a domain user name. If you use a domain user name, log on under a domain with username and password which has administrative privileges on each node.

- c. Remove entries made to the registry by the Oracle Database Configuration, as described in "[Cleaning Up the Registry After an Oracle Database Configuration Assistant Failure on Windows NT](#)" on page C-2.

3. Select the nodes where you want the software installed. You must always select at least two nodes, including the local node. If you deselect the local node, it will still be selected.

The following page appears:

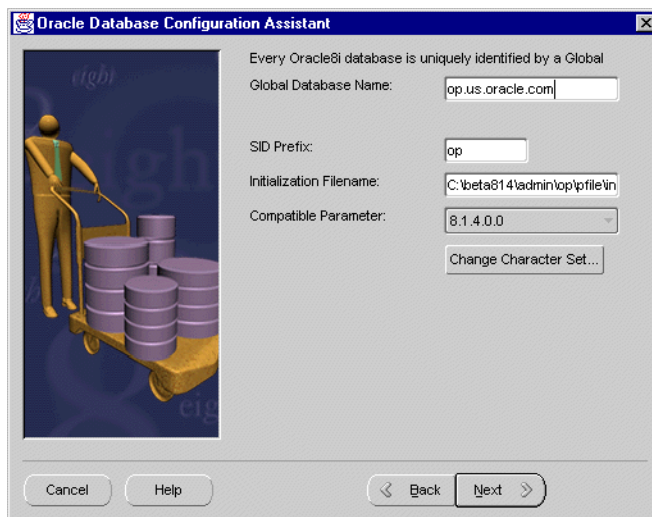


Option	Description
Copy existing database files from the CD	Creates a standard starter database with parameters for each instance in <code>INITSID.ORA</code> and common parameters shared from node-to-node in <code>INITDB_NAME.ORA</code> .
Create new database files	Customizes the creation of the following database types. <ul style="list-style-type: none"> ■ Online Transaction Processing (OLTP) database ■ Decision Support System (DSS) database ■ Hybrid database (combination of OLTP and DSS)

4. Respond to instructions in each Oracle Database Configuration Assistant page, then click Next when you are ready to continue to the next page.

When you get to the page that requests the global database name and SID prefix:

- Enter an appropriate global database name, database name and domain, for the Oracle Parallel Server, such as *OP.domain*.
- Accept or change the prefix for the Oracle System Identifier (SID) that will be used in the SID field. The prefix entered here will be appended with each node's thread ID to create the SID. For example, if *OP* is entered here, the first node would have a SID of *OP1*, the second node *OP2*, and so on.



5. When you get to the last screen, click Finish to start the creation of the Oracle Parallel Server database.

Using Database Creation Assistant results in the creation of the following files:

- *INITDB_NAME.ORA* for general parameters shared by all instances
- *INITSID.ORA* files for each instance in the cluster

See "[Understanding the Initialization Files](#)" on page 6-2 for further information about these files.

Note: If the Oracle Database Creation Assistant fails to create the database, see "[Cleaning Up the Registry After an Oracle Database Configuration Assistant Failure on Windows NT](#)" on page C-2.

Starting the Database in Parallel Mode

To start the Oracle Parallel Server database in parallel mode:

Note: Prior to performing the tasks below, ensure your vendor-supplied CM component is configured and started on each node. See your Operating System Dependent vendor documentation for further information.

1. On Windows NT only, start OracleService*SID* instance on *each* node.
 - From the MS-DOS command line, enter:

```
C:\> net start OracleServiceSID
```

2. Start the listener on each of the nodes. Enter:

```
lsnrctl  
lsnrctl> start [listener_name]
```

where *listener_name* is the name of the listener defined in the LISTENER.ORA file. It is not necessary to identify the listener if you are using the default listener, named LISTENER.

LSNRCTL will display a status message indicating that the listener has started successfully. Check that all expected services for that listener are listed in the services summary in the status message.

3. On the *one of the nodes*, start the database:

```
sql> CONNECT internal/password  
sql> STARTUP;
```

The first instance to start up in shared mode determines the values of the global constant parameters for the other instances. The control file records the values of the GC_* parameters when the first instance starts up.

When another instance attempts to start up in shared mode, the Oracle Server compares the values of the global constant parameters in its parameter file with those already in use and issues a message if any values are incompatible. The instance cannot mount the database unless it has the correct values for its global constant parameters.

4. On the remaining nodes, start the database:

```
sql> CONNECT internal/password  
sql> STARTUP;
```

Verifying Instances Are Running

To verify instances are running:

1. On any node, enter the following:

```
sql> CONNECT internal/password
sql> SELECT * from v$instance;
```

The following output is returned:

```
INST_NUMBER  INST_NAME
-----
1  node1:ops1
2  node2:ops2
3  node3:ops3
. .
```

2. On *all* the nodes, connect as SCOTT/TIGER, and view the EMP table.

```
sql> CONNECT scott/tiger
sql> SELECT * from emp;
```

The employee table displays:

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	19-APR-87	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-87	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

14 rows selected.

Configuring Net8 for Nodes

A user identifies an Oracle Parallel Server instance, its database service and the location of the database on the network through a *net service name*. When an end user connects to an instance, the service is identified by the net service name. The nodes and clients must be configured with net service names.

Services coordinate their sessions with the help of a listener. The listener receives connection requests on behalf of a client application. Listeners are configured to "listen on" an address for a database service. See the *Net8 Administrator's Guide* for further information about net service names and the listener.

Each node must have the following configuration files in `$ORACLE_HOME/network/admin` on UNIX or `ORACLE_HOME\network\admin` on Windows NT:

Configuration File	Description	Configuration Requirements
LISTENER.ORA	Includes addresses of all network listeners on a server, the SIDs of the databases for which they listen, and various control parameters used by the listener.	<p>Note: The LISTENER.ORA file is already configured with the necessary data by the Net8 Configuration Assistant and Oracle Database Configuration Assistant during the installation and configuration process.</p> <p>In an Oracle Parallel Server environment that uses Oracle Enterprise Manager, the LISTENER.ORA file on <i>each</i> node must be configured with:</p> <ul style="list-style-type: none"> ■ a TCP/IP address ■ an entry for the SID of the node in the <code>SID_LIST_listener_name</code> section. For example: <pre>SID_LIST_LISTENER = (SID_LIST = (SID_DESC = (SID_NAME = op1) (GLOBAL_DBNAME = op.com)))</pre>

Configuration File	Description	Configuration Requirements
TNSNAMES.ORA	<p>Includes a list of network descriptions of service names, called net service names.</p> <p>Note: While net service names may be stored in various ways, Oracle Corporation recommends using a TNSNAMES.ORA file. In order to use the Oracle Enterprise Manager products, a TNSNAMES.ORA file is required.</p>	<ul style="list-style-type: none"> ■ A net service name must be specified for each node's instance. The service name specified must be the same for all nodes, and the instance name specified must be unique for each instance. ■ A net service name should be specified for the database.
SQLNET.ORA	Includes the names resolution method.	Because the net service names are specified in TNSNAMES.ORA files, the SQLNET.ORA must specify that TNSNAMES.ORA be used when resolving a net service name.

Note: Oracle Enterprise Manager uses entries from TNSNAMES.ORA files on the managed nodes. If your network has Oracle Names servers, TNSNAMES.ORA files must still be configured.

The steps to perform to configure the network include:

[Step 1: Configure the Listener](#)

[Step 2: Configure Net Service Names](#)

[Step 3: Configure SQLNET.ORA](#)

[Configuring Net8 for Clients](#)

Step 1: Configure the Listener

Listener configuration should not be necessary, as it is taken care of as a part of the installation process, as described in:

- [Typical Installation Results](#)
- [Custom Installation Results](#)

If configuration is still necessary, follow the instructions in "[Configuring LISTENER.ORA](#)" on page 4-16.

Typical Installation Results

During a Typical installation, the Net8 Configuration Assistant creates a default LISTENER.ORA file for a listener named LISTENER that listens on the following protocol addresses:

- IPC for connections to external procedures
- TCP/IP for a Net8 client connection to the Oracle8i database
- TCP/IP for a Java client connection to the Java option

Additionally, the Oracle Database Configuration Assistant configures information about the database, including the global database name and SID in the LISTENER.ORA. Typically, static configuration of this information is not necessary in a release 8.1 configuration, because instances register this information with the listener upon startup. However, this information is necessary for Oracle Intelligent Agent to work properly.

A sample LISTENER.ORA is shown below:

```
listener =
  (description_list =
    # IPC listenening endpoint for external procedures' connections
    (description =
      (address_list =
        (address = (protocol = ipc)(key = extproc0))
      )
    )
    # TCP/IP listenening endpoint for java option connections
    (description =
      (protocol_stack =
        (presentation = giop)
        (session = raw)
      )
      (address_list =
        (address = (protocol = tcp)(host = opshp1-pc)(port = 2481))
      )
    )
    # TCP/IP listenening endpoint for database connections
    (description =
      (address_list =
        (address = (protocol = tcp)(host = opshp1-pc)(port = 1521))
      )
    )
  )
)
```



```
sid_list_listener =
  (sid_list =
    (sid_desc =
      (sid_name = plsextproc)
      (oracle_home = c:\orahome)
      (program = extproc)
    )
  # Information about database needed for Oracle Intelligent Agent
  (sid_desc =
    (global_dbname = op.us.oracle.com)
    (oracle_home = c:\orahome\ops)
    (sid_name = opl)
  )
)
```

Custom Installation Results

During a Custom installation, a LISTENER.ORA file should have been created for a listener named by you for a protocol address you configured, as described in ["Custom Installation"](#) on page 3-6.

Additionally, the Oracle Database Configuration Assistant configures information about the database, including the global database name and SID in LISTENER.ORA. Typically, static configuration of this information is not necessary in a release 8.1 configuration, because instances register this information with the listener upon startup. However, this information is necessary for Oracle Enterprise Manager version 2 to work properly.

Configuring LISTENER.ORA

If the LISTENER.ORA is not configured for some reason, use the Net8 Assistant to perform the following procedure on *each* node:

1. Stop the listener. At the command line, enter:

```
lsnrctl  
lsnrctl> stop [listener_name]
```

listener_name is the name of the listener defined in the LISTENER.ORA file. It is not necessary to identify the listener if for the default listener, named LISTENER.

2. Start the Net8 Assistant:

-On UNIX, run the shell script `netasst` at `$ORACLE_HOME/bin`.

-On Windows NT, choose Start > Programs > Oracle - *HOME_NAME* > Network Administration > Net8 Assistant.

3. Double-click on the Listeners folder.

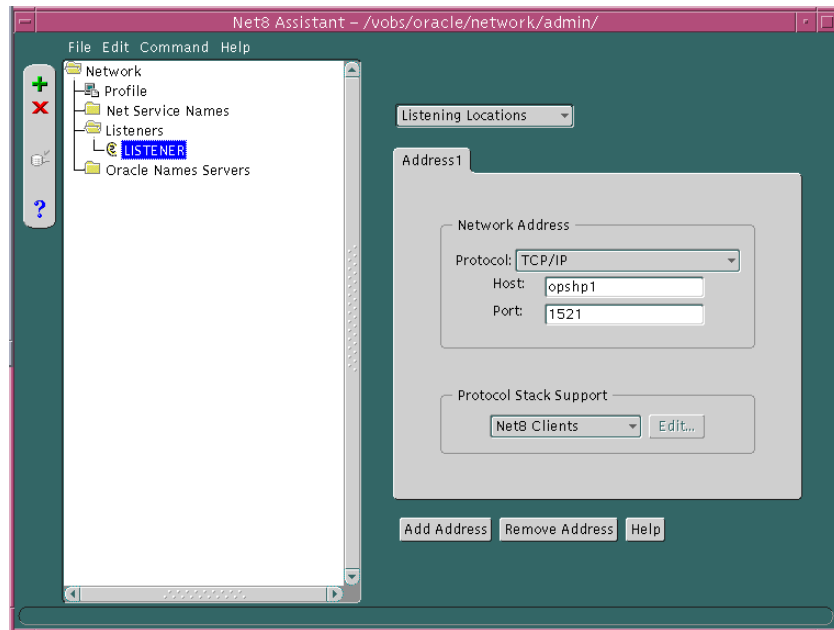
4. Select the listener. If one does not exist:

- a. Select Create from the Edit menu or click on the "+" button to create a new listener. The *Choose Listener Name* dialog appears.

- b. Enter the listener name, then click OK. This listener is added.

5. Choose Listening Locations from the drop-down list box.

6. Click Add Address:

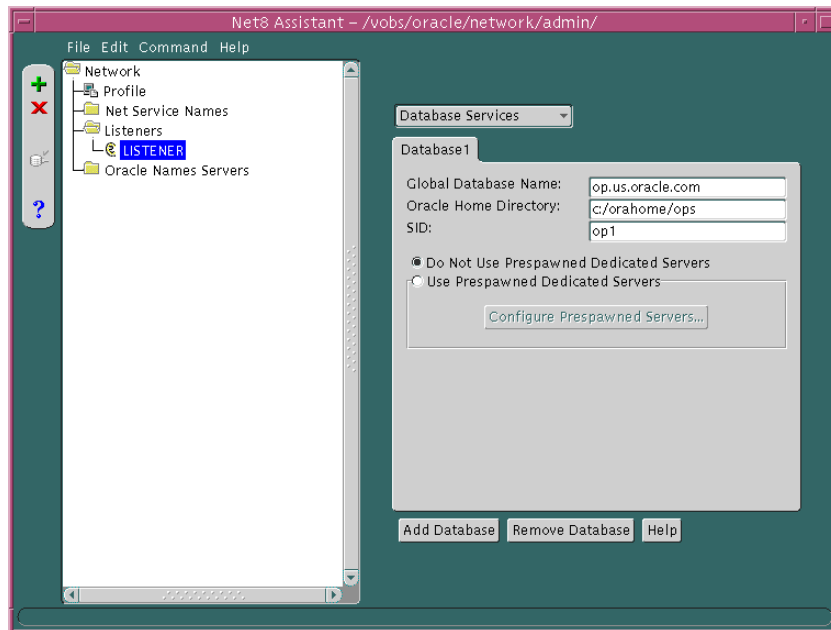


7. Choose TCP/IP as the protocol, enter the name of the node's host, and choose port 1521.

Note: If you do not choose the registered port of 1521, you must configure the LOCAL_LISTENER parameter in the INITDB_NAME.ORA file and resolve it with a net service name entry in the TNSNAMES.ORA file. For further information about configuring non-default port numbers in LISTENER.ORA, see Chapter 6, "Configuring Naming Methods and the Listener" in the *Net8 Administrator's Guide*.

8. Choose Database Services from the drop-down list box.

9. Click Add Database:



10. Enter the global database name for the database in the Global Database Name field, the Oracle home in the Oracle Home Directory field, and the SID of the node's instance in the SID field.
11. Select Save Network Configuration from the File menu.
12. Choose Exit from the File menu to exit the Net8 Assistant application.
13. Start the listener. At the command line, enter:

```
lsnrctl
lsnrctl> start listener_name
```

listener_name is the name of the listener defined in the LISTENER.ORA file. It is not necessary to identify the listener if for the default listener, named LISTENER.

Note: On UNIX platforms, to use a single LISTENER.ORA file that lists all listeners on all nodes, the listeners must have distinct names. This can be accomplished by setting the `lsnr_listener_name` parameter in the `DB_NAME.CONF` file, located in `$ORACLE_HOME/ops`:

```
lsnr_listener_name = "listener_%m"
```

The LISTENER.ORA file can then be distributed to all the nodes.

For further information about the `DB_NAME.CONF` file, see ["Parameter Descriptions"](#) on page B-5.

Step 2: Configure Net Service Names

A net service name is mapped to a network address contained in a connect descriptor in a TNSNAMES.ORA file. A *connect descriptor* contains the location of the listener through a protocol address and the service name of the service to which to connect. Clients use this net service name when making a connection with an application.

This section discusses the following:

- [Creating Entries in TNSNAMES.ORA for Each Instance](#)
- [Creating An Entry in the TNSNAMES.ORA File for the Database and Enabling Client Load Balancing](#)
- [Additional Configuration Required for Windows NT in the TNSNAMES.ORA File](#)

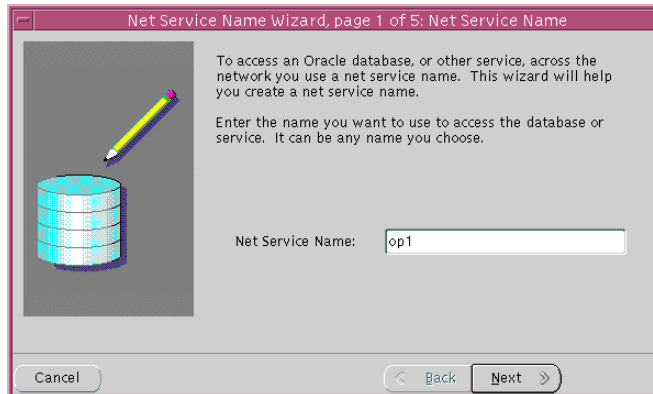
Creating Entries in TNSNAMES.ORA for Each Instance

A net service name *must* be created for each Oracle Parallel Server instance.

To configure a TNSNAMES.ORA file for an Oracle Parallel Server, use the Net8 Assistant:

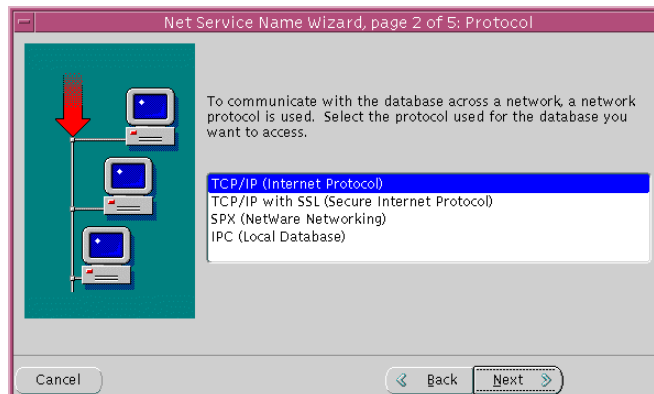
1. On one of the nodes, start the Net8 Assistant:
 - On UNIX, run the shell script `netasst` at `$ORACLE_HOME/bin`.
 - On Windows NT, choose Start > Programs > Oracle - *HOME_NAME* > Network Administration > Net8 Assistant.
2. Double-click on the Net Service Name folder in the directory tree.
3. Select Create from the Edit menu or click on the "+" button to create a new net service name. The *Net Service Name Wizard* starts.
4. Enter a name in the New Net Service Name field, and click Next.

For clarity, the name should reflect the SID of instance on the node, such as OP1, OP2, and so on.

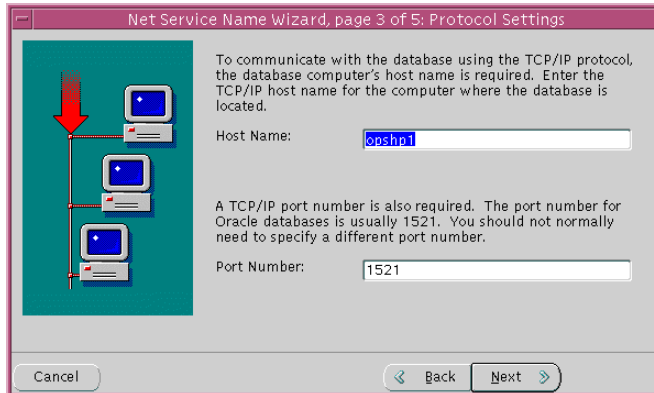


If you want to be explicit, the net service name may be qualified with the domain.

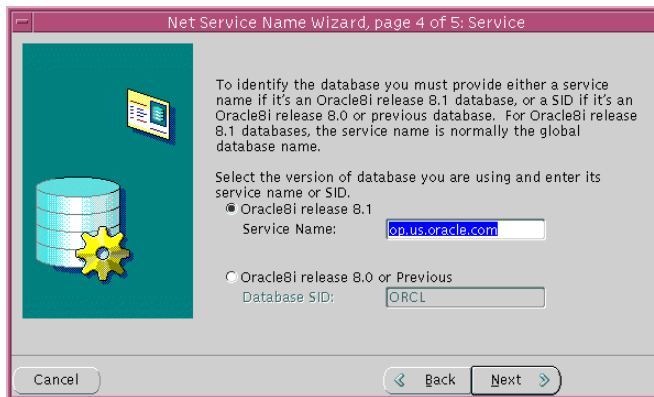
5. Select the protocol the listener is configured to listen on (this protocol must also be installed on the clients), then click Next:



6. Enter the appropriate listener protocol address information for your chosen protocol in the fields, then click Next:



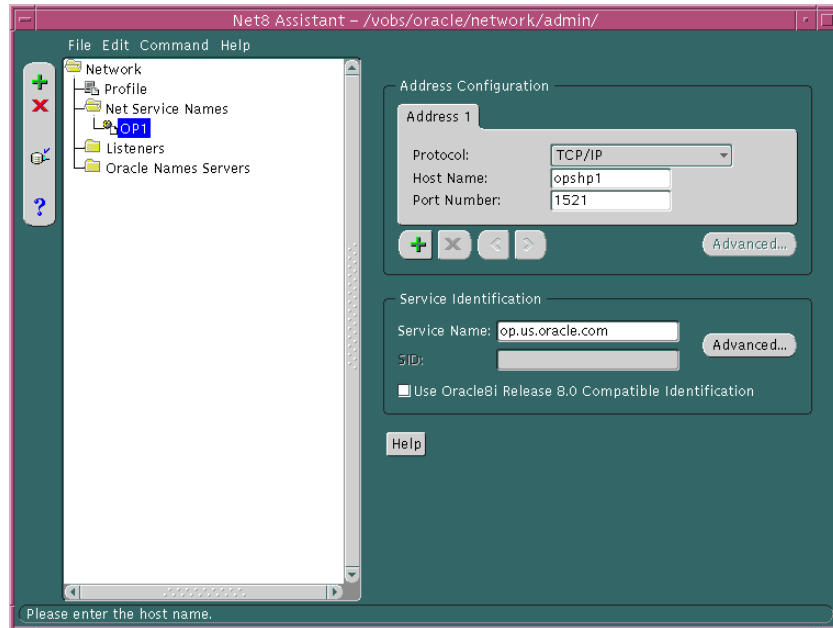
7. Click Oracle8 Version 8.1, enter a service name in the Service Name field, then click Next:



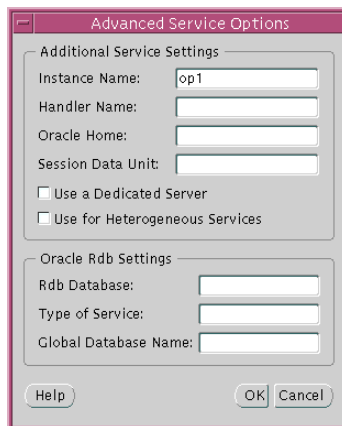
The service name is typically the global database name, a name comprised of the database name and domain, entered during installation. An example service name is OP.US.ORACLE.COM, where OP is the database name and US.ORACLE.COM is the domain.

8. Click Finish

The net service name is added to the Net Service Names folder:



9. Click **Advanced** in the **Service Identification** group box. The *Advanced Service Options* dialog box appears:



10. Enter the instance name in the Instance Name field, then click OK. The instance name value may be obtained from the INSTANCE_NAME value in INITSID.ORA

The instance name should match the net service name or SID of the instance on the node, such as OP1, OP2, and so on.

11. Repeat Steps 2-10 for the other net service names. The value of the service name should be identical for all instances, but the instance names should be unique.
12. Choose Save Network Configuration from the File menu.
13. Choose Exit from the File menu to exit the Net8 Assistant.
14. Copy the newly created TNSNAMES.ORA configuration file to the other nodes and clients. This will ensure the files are consistent. Otherwise, you must use Net8 Assistant on every client, introducing possible errors.

A sample TNSNAMES.ORA file is shown below:

```
op1 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op1))
op2 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op2))
```

Creating An Entry in the TNSNAMES.ORA File for the Database and Enabling Client Load Balancing

Oracle Corporation recommends creating a net service name for the Oracle Parallel Server database. Optionally, this net service name can also be created so that it randomly selects instances with client load balancing.

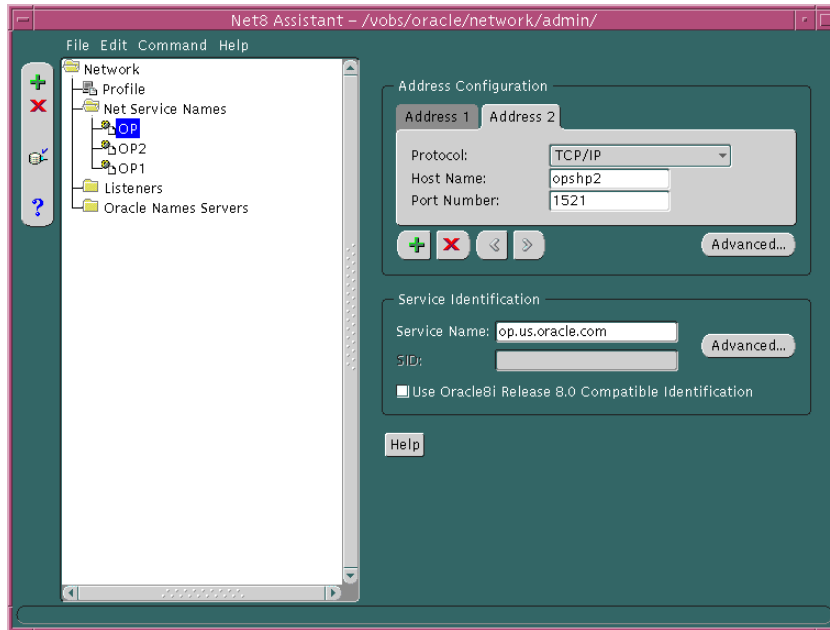
Note: Besides client load balancing, this net service name can also be set up for connect-time failover and transparent application failover. Configuring both types of failover requires additional configuration to LISTENER.ORA and TNSNAMES.ORA, as described in "[Configuring Net8 with Failover](#)" on page 4-35.

Client load balancing (LOAD_BALANCE = ON) instructs Net8 to balance the load over the list of listener addresses by picking one at random. Client load balancing is turned ON by default for multiple connect descriptors (DESCRIPTION_LIST).

To create a net service name for the database and enable client load balancing:

1. Create a new net service name, following steps 1-7 in "[Creating Entries in TNSNAMES.ORA for Each Instance](#)" on page 4-20. Instead of naming the net service name after the node's instance, name it after the database, for example, OP, and configure the address for one of the nodes:
2. Double-click the Net Service Names folder, and click the newly-created net service name.

3. Click the "+" button:

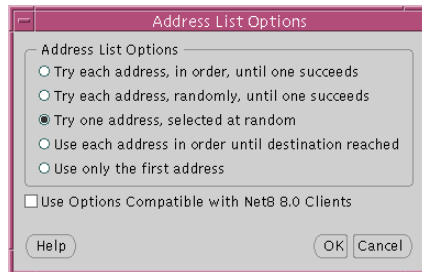


4. Enter the address information for the next node.
5. Repeat steps 3-4 for each additional node.

Addresses created in this manner will be tried by the node in the order presented. To randomly have the node choose between the multiple address during a connection:

- a. Click Advanced in the Address Configuration group.

The *Address List Options* dialog box appears:



- b.** Click *Try one address, selected at random*, then click OK.
- 6.** Choose Save Network Configuration from the File menu.
The TNSNAMES.ORA file is re-created.
- 7.** Choose Exit from the File menu.
The Net8 Assistant application exits.

Shown below is a 8.1 release TNSNAMES.ORA file with net service names of OP, OP1 and OP2. OP allows the node to connect to the database service OP and try one address at random. OP1 and OP2 allow the node to connect to a specific Oracle Parallel Server instance.

```
op =
  (description=
    (load_balance=on)
    (address=
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (address=
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data=
      (service_name=op.us.oracle.com)))
op1 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op1))
op2 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op2))
```

Additional Configuration Required for Windows NT in the TNSNAMES.ORA File

If you plan to use Oracle Enterprise Manager and OPS is installed on a Windows NT cluster, additional net service names must be added to the TNSNAMES.ORA file.

When Oracle Enterprise Manager performs a startup of instances running on UNIX nodes, a command is passed to a process on the node to perform a local connection.

On Windows NT, a remote connection is performed from the one of the nodes to the other nodes using entries in the TNSNAMES.ORA file. This remote connection cannot accept the SERVICE_NAME and INSTANCE_NAME parameters in the TNSNAMES.ORA if the database is down. This occurs because this information about the database is not registered with the listener unless the database is up.

The workaround to this problem is to specify additional net service names that use SID rather than SERVICE_NAME and INSTANCE_NAME. If you plan to use Oracle Enterprise Manager and Oracle Parallel Server is installed on Windows NT nodes, you must perform the following:

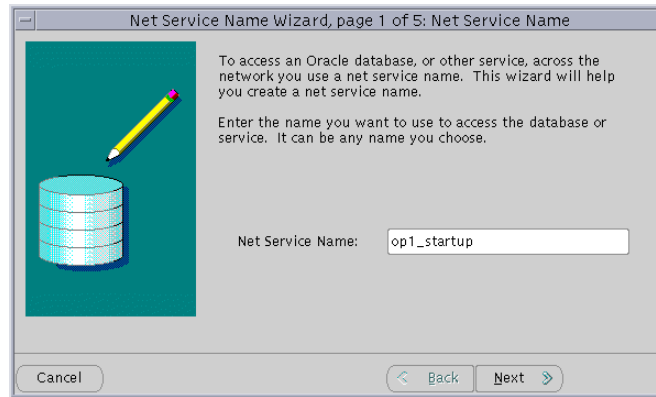
- Create net service names that use a format of *SID_startup*
When Oracle Enterprise Manager performs a startup of an instance, it will expect this name.
- Ensure the CONNECT_DATA portion of the net service name uses SID rather than SERVICE_NAME
- Ensure the *SID_startup* net service names follow in sequence after the other net service names. Oracle Enterprise Manager uses net service names to determine the name of the instance. If the *SID_startup* entries are at the top of the list, instances will be named *SID_startup* rather than *SID*. This discrepancy between what is discovered and the actual instance names will prevent you from starting or stopping instances.

To configure the additional net service names on Windows NT nodes:

1. Create net service names for each node's instances, as described in ["Creating Entries in TNSNAMES.ORA for Each Instance"](#) on page 4-20 and ["Creating An Entry in the TNSNAMES.ORA File for the Database and Enabling Client Load Balancing"](#) on page 4-25.
2. Create additional net service names named *SID_startup* for each node's instance using SID rather than SERVICE_NAME and INSTANCE_NAME.
 - a. In the Net8 Assistant, double-click on the Net Service Name folder in the directory tree.

- b. Select Create from the Edit menu or click on the "+" button to create a new net service name. The *Net Service Name Wizard* starts.
- c. Enter a name in the New Net Service Name field, and click Next.

The name should be *SID_startup*, such as *OP1_startup*:



- d. Select the network protocol you want to use (this protocol must also be installed on the clients), and click Next.
- e. Enter the appropriate listener address information for your chosen protocol in the fields, and click Next.
- f. Click *Oracle8 Version 8.0 or Previous*, enter the SID in the Database SID field, such as *OP1*, and click Next.

The net service name is added to the Net8 Service Names folders.

- g. Repeat Steps a-f for each node's instance in the cluster.
- h. Choose Save Network Configuration from the File menu.
- i. Choose Exit from the File menu to exit the Net8 Assistant.
- j. Copy the newly created TNSNAMES.ORA configuration file to the other nodes and clients. This will ensure the files are consistent. Otherwise, you must use Net8 Assistant on every client, introducing possible errors.

A sample TNSNAMES.ORA file is shown below:

```
op =
  (description=
    (load_balance=on)
    (address=
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (address=
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data=
      (service_name=op.us.oracle.com)))
op1 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op1))
op2 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op2))
op1_startup =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data =
      (sid = op1))
```

```
op2_startup =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data =
      (sid = op2))
```

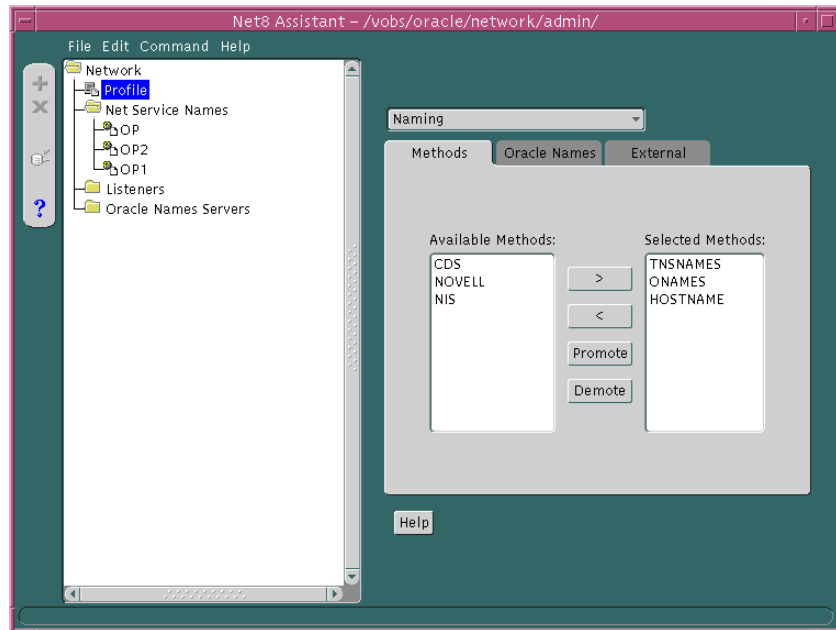
Step 3: Configure SQLNET.ORA

Configure local naming as the first method specified in the NAMES_DIRECTORY_PATH parameter in the SQLNET.ORA file. This parameter specifies the order of naming methods Net8 will use to reconcile services.

To specify local naming as the first naming method:

1. Start the Net8 Assistant:
 - On UNIX, run `netasst` at `$ORACLE_HOME/bin`.
 - On Windows NT, choose Start > Programs > Oracle - *HOME_NAME* > Network Administration > Net8 Assistant.
2. Click the Profile icon.
3. Choose Naming from the drop-down list box.
4. Click the Methods tab.
5. Choose TNSNAMES from the Available Methods list, then click ">".

6. Select TNSNAMES in the Selected Methods list, then use the Promote button to move the selection to the top of the list:



7. Choose Save Network Configuration from the File menu.
8. Choose Exit from the File menu to exit the Net8 Assistant application.

The SQLNET.ORA file should contain an entry that lists TNSNAMES first in the NAMES.DIRECTORY_PATH parameter:

```
names.directory_path = (tnsnames, onames, hostname)
```

9. Copy the newly created SQLNET.ORA configuration file to the other nodes and clients. This will ensure the files are consistent. Otherwise, you must use Net8 Assistant on every client, introducing possible errors.

Configuring Net8 for Clients

Clients should be configured with a TNSNAMES.ORA and SQLNET.ORA files, as described in

["Step 2: Configure Net Service Names"](#) on page 4-20

["Step 3: Configure SQLNET.ORA"](#) on page 4-32

If the client is running Oracle Enterprise Manager or Oracle Performance Manager, see [Chapter 5, "Installing and Configuring Oracle Parallel Server Management"](#) for further information about configuring Net8 for these applications.

Configuring Net8 with Failover

You can enable the following types of failover:

- Connect-Time Failover (FAILOVER = ON)
- Transparent Application Failover (FAILOVER_MODE)

Connect-time failover instructs Net8 to fail over to a different listener if the first listener fails when set to ON. The number of addresses in the list determines how many addresses are tried. When set to OFF, instructs Net8 to try one address.

Failover is turned ON by default for multiple address lists (ADDRESS_LIST), connect descriptors (DESCRIPTION), and multiple connect descriptors (DESCRIPTION_LIST).

Transparent application failover instructs Net8 to fail over to a different listener if the first listener fails during runtime. Depending upon the configuration, session or any SELECT statements which were in progress are automatically failed over. This parameter must be embedded under CONNECT_DATA.

See the *Net8 Administrator's Guide* for further conceptual information about connect-time failover and *Oracle8i Tuning* for further conceptual information about transparent application failover.

Establishing both kinds of failover requires that you:

- [Configure the Listener](#)
- [Configuring the TNSNAMES.ORA File for Failover](#)

Configure the Listener

Implementing failover does not allow use of static service configuration parameters in the LISTENER.ORA file, such as:

```
sid_list_listener =  
  (sid_desc =  
    (global_dbname = op.us.oracle.com)  
    (oracle_home = c:\orahome\ops)  
    (sid_name = op1)  
  )  
)
```

However, this information is the default implementation and is required by the Oracle Intelligent Agent for Oracle Enterprise Manager.

In order to implement failover, an additional listener must be specified on each node in the LISTENER.ORA file without service registration information. Perform the following steps:

Step 1: Stop the Listener

Step 2: Add Entry for New Listener in LISTENER.ORA

Step 3: Start the Listener

Step 1: Stop the Listener

Stop the listener. At the command line, enter:

```
lsnrctl  
lsnrctl> stop [listener_name]
```

listener_name is the name of the listener defined in the LISTENER.ORA file. It is not necessary to identify the listener if you are using the default listener, named LISTENER.

Step 2: Add Entry for New Listener in LISTENER.ORA

Add an entry for a new listener to the LISTENER.ORA file that listens on TCP/IP protocol with the same port number as the first listener, which is typically listening on port 1521. This information must be added manually, as the Net8 Assistant does not permit using the same port number for another listener.

Do not add a service information with the `SID_LIST_listener_name` parameter.

Note: If you do not use the registered port of 1521, you must configure the `LOCAL_LISTENER` parameter in the `INITDB_NAME.ORA` file and resolve it with a net service name entry in the `TNSNAMES.ORA`. For further information about configuring non-default port numbers in LISTENER.ORA, see Chapter 6, "Configuring Naming Methods and the Listener" in the *Net8 Administrator's Guide*.

For example, a second listener named LISTENER1 has been added to the LISTENER.ORA file below:

```
# listening endpoints for a listener named LISTENER1
listener1 =
  # TCP/IP listenening endpoint for database connections
  (description =
    (address = (protocol = tcp)(host = opshp1)(port = 1521))
  )
)
listener =
  (description =
    (address = (protocol = tcp)(host = opshp1)(port = 1521))
  )
)
# Static information about services for a listener named LISTENER
sid_list_listener =
  (sid_list =
    # Database information needed for Oracle Intelligent Agent
    (sid_desc =
      (global_dbname = op.us.oracle.com)
      (oracle_home = c:\orahome\ops)
      (sid_name = op1)
    )
  )
)
```


Step 3: Start the Listener

Start only the newly configured listener. At the command line, enter:

```
lsnrctl  
lsnrctl> start listener_name
```

listener_name is the name of the new listener defined in the LISTENER.ORA file (LISTENER1).

Configuring the TNSNAMES.ORA File for Failover

Failover is configured in the TNSNAMES.ORA file:

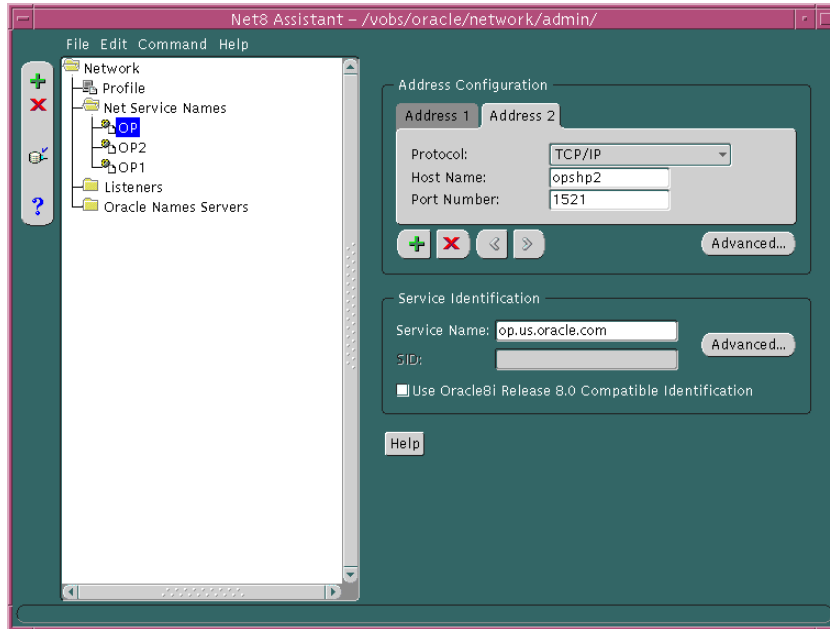
- [Configuring Connect-Time Failover](#)
- [Configuring Transparent Application Failover](#)

Configuring Connect-Time Failover

To configure connect-time failover for a client:

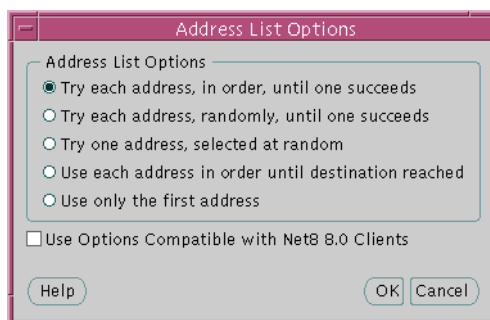
1. Create a new net service name, following steps 1-7 in "[Creating Entries in TNSNAMES.ORA for Each Instance](#)" on page 4-20. Instead of naming the net service name after the node's instance, name it after the database, for example, OP, and configure the address for one of the nodes:
2. Double-click the Net Service Names folder, and click the newly-created net service name.

3. Click the "+" button:



4. Enter the address information for the next node.
5. Repeat steps 3-4 for each additional node.
6. Click Advanced in the Address Configuration group.

The *Address List Options* dialog box appears:



7. Ensure the *Use Options Compatible with Net8 8.0 Clients* option is OFF.
8. Use one of the connect-time failover options indicated below, then click OK.

Net8 Assistant Option	TNSNAMES.ORA Parameter Setting
<i>Try each address, in order, until one succeeds</i>	FAILOVER=ON
<i>Try each address, randomly, until one succeeds</i>	LOAD_BALANCE=ON
This option turns both connect-time failover and client load balancing to ON.	FAILOVER=ON

9. Choose Save Network Configuration from the File menu.
The TNSNAMES.ORA file is re-created.
10. Choose Exit from the File menu.
The Net8 Assistant application exits.

Shown below is a 8.1 release TNSNAMES.ORA file with net service names of OP, OP1 and OP2. OP allows the node to connect to the database service OP and try each address at random until one succeeds. OP1 and OP2 allow the node to connect to a specific Oracle Parallel Server instance.

```
op =
  (description=
    (load_balance=on)
    (failover=on)
    (address=
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (address=
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data=
      (service_name=op.us.oracle.com)))
op1 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op1))
op2 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op2))
```

Configuring Transparent Application Failover

Transparent application failover involves manual configuration that includes specification of a primary node with a typical ADDRESS parameter and a backup node, failover type and failover method with a FAILOVER_MODE parameter.

FAILOVER_MODE can be implemented with connect-time failover with multiple addresses. In the example below, Net8 will try to make a connection to the one of the listener address (OPSHP1 or OPSHP2) at random. If, after the connection, the node should go down, Net8 will fail over to the other node, reserving any SELECT statements in progress.

```
op =
  (description=
    (load_balance=on)
    (failover=on)
    (address=
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (address=
      (protocol = tcp)
      (host = opshp2)
      (port = 1521))
    (connect_data=
      (service_name=op.us.oracle.com)
      (failover_mode =
        (type = select))
    )
  )
```

A specific address can be specified to initially make a connection. Additionally, a backup node can be specified in the `FAILOVER_MODE` parameter in case of a failure. In the example below, there is no connect-time failover. Net8 will try only OPSHP1. If, after the connection, OPSHP1 should go down, Net8 will fail over to OPSHP2, reserving any `SELECT` statements in progress.

```
op =
  (description=
    (address=
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data=
      (service_name=op.us.oracle.com)
      (failover_mode =
        (backup = opshp2)
        (type = select)
        (method = basic)
      )
    )
  )
)
```

`FAILOVER_MODE` parameters are described below:

PARAMETER	Description
BACKUP	Specifies the failover node.
TYPE (Required)	<p>Specifies the type of failover. Three types of Net8 failover functionality are available by default to Oracle Call Interface (OCI) applications:</p> <p>SESSION: Fails over the session; that is, if a user's connection is lost, a new session is automatically created for the user on the backup. This type of failover does not attempt to recover selects.</p> <p>SELECT: Allows users with open cursors to continue fetching on them after failure. However, this mode involves overhead on the client side in normal select operations.</p> <p>NONE: This is the default, in which no failover functionality is used. This can also be explicitly specified to prevent failover from happening.</p>

PARAMETER	Description
METHOD	<p>Determines how fast failover occurs from the primary node to the backup node:</p> <p>BASIC: Establishes connections at failover time. This option requires almost no work on the backup server until failover time.</p> <p>PRECONNECT: Pre-establishes connections. This provides faster failover but requires that the backup instance be able to support all connections from every supported instance</p>

Testing the Net8 Configuration

To ensure the files are configured correctly:

1. On any node or client machine, connect to an instance:

```
sql> CONNECT internal/password@net_service_name
```

A "*Connected*" message should be returned to the screen.

If there is an error in connecting, you must troubleshoot your installation. Typically, there is a problem with the IP address, host name, service name, or instance name.

2. On the one node, increase MILLER's salary by \$1000 and commit the change:

```
sql> UPDATE emp
set sal = sal + 1000
where ename = 'miller';
sql> commit;
```

3. On the other nodes, select the EMP table again:

```
sql> SELECT * from emp;
```

MILLER's salary should now be \$2,300, indicating all the instances can see the database.

Installing and Configuring Oracle Parallel Server Management

Oracle Enterprise Manager supports administration of Oracle Parallel Server databases with Oracle Parallel Server Management. Oracle Parallel Server Management enables certain parallel components of Oracle Enterprise Manager. A performance monitoring tool called Oracle Performance Manager further enhances Oracle Parallel Server Management, allowing end users to monitor the global VS view tables for performance.

This chapter describes how to install and configure Oracle Parallel Server Management.

Specific topics discussed are:

- [Oracle Parallel Server Management Architecture](#)
- [Oracle Parallel Server Requirements](#)
- [Understanding Oracle Enterprise Manager Setup](#)
- [Installing Oracle Enterprise Manager](#)
- [Configuring Oracle Enterprise Manager](#)
- [Installing Oracle Performance Manager](#)
- [Configuring Oracle Performance Manager](#)

Oracle Parallel Server Management Architecture

You can control the activity of Oracle Parallel Servers and their instances using Oracle Parallel Server Management. Oracle Parallel Server Management is a comprehensive and integrated system management solution for the Oracle Parallel Server. Oracle Parallel Server Management allows you to manage multi-instance databases running in heterogeneous environments through an open client-server architecture.

In addition to managing parallel databases, Oracle Parallel Server Management allows you to schedule jobs, perform event management, monitor performance, and obtain statistics to tune parallel databases.

Oracle Enterprise Manager provides database administrators (DBAs) with a powerful set of tools to manage, monitor, and administer even the most complex network of databases from a single workstation, called the Enterprise Manager Console.

Oracle Parallel Server Management within the Oracle Enterprise Manager context, is comprised of the following components:

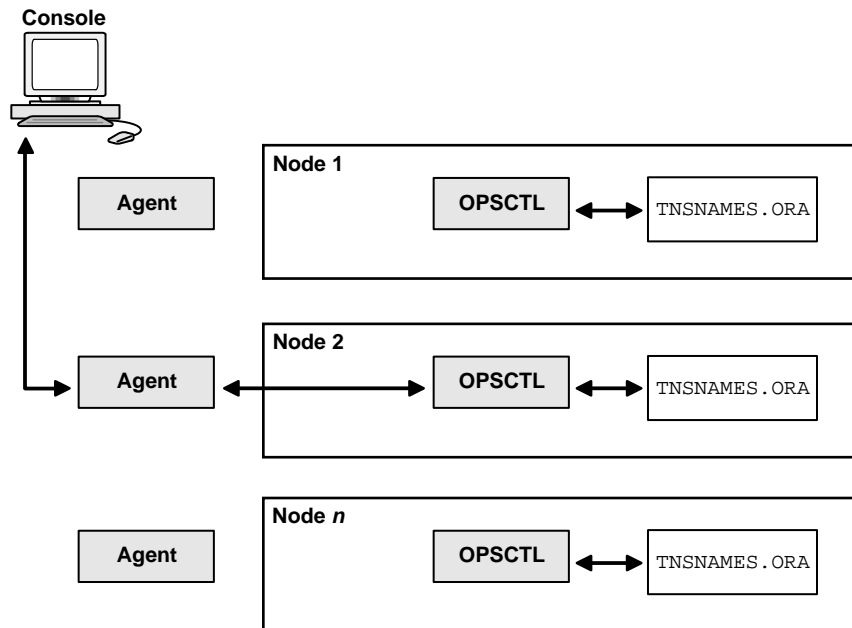
Component	Description
Console	<p>The Console is a graphical user interface that provides menus, toolbars, and launch palettes to allow access to Oracle tools. The Console is run from a client machine or a browser.</p> <p>The Console is comprised of four windows:</p> <ul style="list-style-type: none">■ Navigator window — Contains an object browser which provides an organized, consistent and hierarchical view of the database objects in the network.■ Group window — Provides a customized, graphical representation of key objects, created by the administrator.■ Event Management window — Allows the administrator to remotely monitor for critical database and system events.■ Job window — Provides the administrator with the ability to automate repetitive activities.
Management Server	<p>The Management Server performs functions requested by the Console. Each Management Server manages exactly one domain. The domain is the logical grouping of machines for administrative purposes.</p> <p>The Management Server processes all system management tasks and administers the distribution of these tasks to the Oracle Intelligent Agents on the managed nodes.</p>

Component	Description
Repository Database	<p>All administrators in a domain share a repository, a set of database tables. The repository contains the accounts of every administrator in the domain.</p> <p>The Management Server uses the repository to store all system data, application data, and the state of managed nodes throughout the environment. The repository can be shared by multiple administrators.</p> <p>The repository <i>must</i> be on a separate machine from the nodes.</p>
Oracle Intelligent Agents	<p>Manages and completes tasks requested from the Console. Once installed, the Oracle Intelligent Agent:</p> <ul style="list-style-type: none"> <li data-bbox="632 595 1329 647">■ Listens for and responds to job requests initiated by the Console. <li data-bbox="632 664 1329 769">■ Schedules job requests. Requests can include detecting and correcting abnormal conditions, performing standard database administration procedures, and monitoring events. <p>The Oracle Intelligent Agent is installed on the nodes.</p>
(Optional) Oracle Performance Manager	<p>The Oracle Performance Manager allows you to choose from a variety of tabular and graphic performance statistics for Oracle Parallel Servers. The statistics represent the aggregate performance for all instances running on an Oracle Parallel Server. The statistics are displayed in individual charts and include information about data block pings, lock activity, file I/O, and session and user information.</p>
(Optional) Oracle Data Gatherer	<p>Gathers performance statistics for the Oracle Performance Manager.</p> <p>The Oracle Data Gatherer must be installed on a node somewhere on the network.</p>

Oracle Parallel Server Management uses the following utilities to manage instances:

Component	Description
OPSCCTL	<p>Serves as a single point of control between the Oracle Intelligent Agent and the nodes.</p> <p>Only one node's Oracle Intelligent Agent is used to communicate to OPSCCTL. OPSCCTL on that node then communicates to the other nodes through Net8.</p> <p>OPSCCTL is installed on the nodes.</p>
OPS Communication Daemon (OPSD)	<p>Receives requests from OPSCCTL to execute administrative job tasks, such as startup or shutdown. The command is executed locally on each node, and the results are sent back to OPSCCTL.</p> <p>OPSD is installed on the nodes. OPSD is only implemented on UNIX platforms.</p>

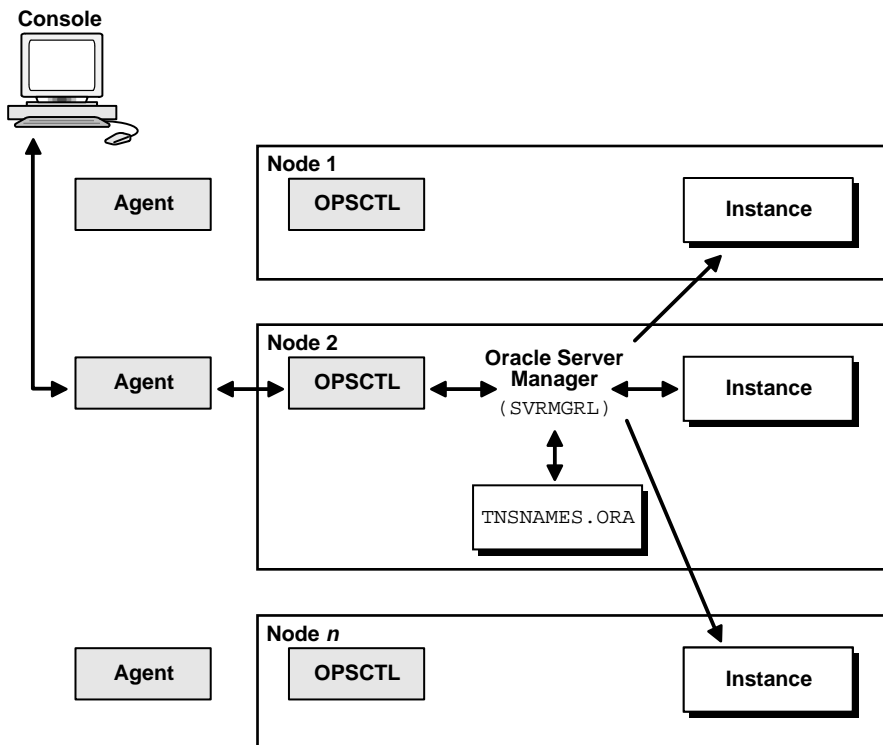
On UNIX, the Oracle Intelligent Agent invokes OPSCCTL to executes jobs. OPSD then receives requests from OPSCCTL, as shown in [Figure 5-1](#):

Figure 5–1 *OPCTL Architecture for UNIX*

On Windows NT, the Oracle Intelligent Agent invokes OPSCTL to execute jobs. OPSCTL then sends requests to Oracle Server Manager. Oracle Server Manager uses information from configured TNSNAMES.ORA files on the nodes to connect to a listener, which sends the OPSCTL requested action to the instance. Therefore, it is essential to have the TNSNAMES.ORA file configured on all nodes in cluster.

Figure 5-2 shows how OPSCTL operates between the agent and the nodes on Windows NT:

Figure 5-2 OPSCTL Architecture for Windows NT



Oracle Parallel Server Requirements

The following requirements must be met in order to use OPSM from the Console:

The requirements for each node are:

Node Requirements	For additional information, see:
Ensure Oracle8i Enterprise Edition, Oracle Parallel Server Option, and Oracle Intelligent Agent are installed on each node in the cluster.	Chapter 3, "Installation"
Each node must have a TNSNAMES.ORA configured. Create a TNSNAMES.ORA file so that each instance has a unique net service name. The net service names should contain the same service name but the instance names should be unique. The LISTENER.ORA file must be configured with information about the database on each node.	"Configuring Net8 for Nodes" on page 4-12
Each instance's Oracle System Identifiers (SIDs) in the Oracle Parallel Server must be unique.	"Unique Identification of Instances" on page 1-11

Understanding Oracle Enterprise Manager Setup

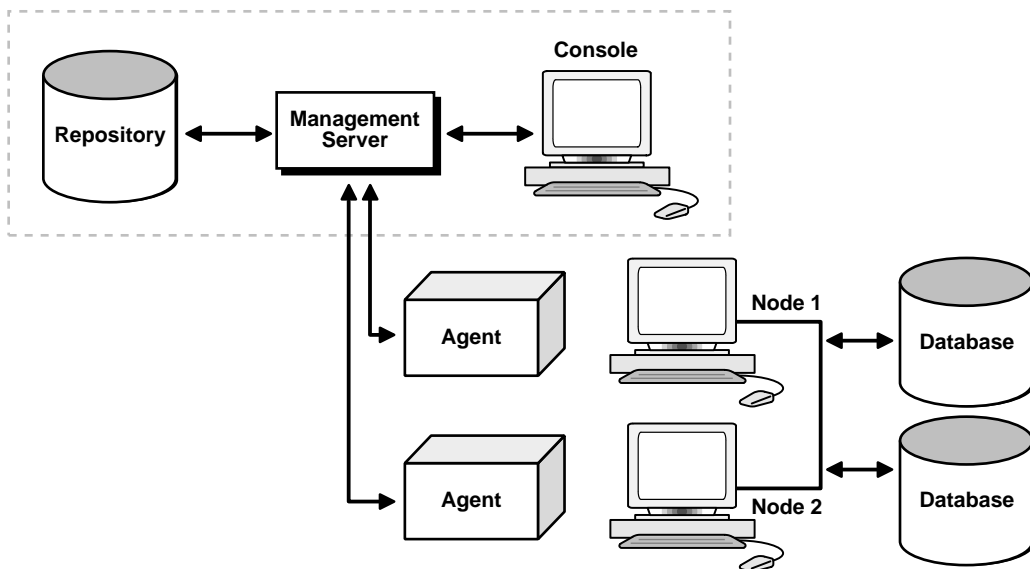
You can run the individual Oracle Enterprise Manager components on separate machines or combine different components on separate machines to collaboratively manage the complete Oracle environment. Two environments are shown below:

Environment 1: Console, Management Server and Repository on same machine

The Console with the aid of the Management Server remotely manages the databases for both nodes. The Console and the Management Server are running on a Windows NT or Solaris machine with an Oracle8 database installed that is only being used as a repository. The Oracle Intelligent Agent does *not* need to be running.

The nodes share an Oracle8*i* database. The repository is *not* created in this database, and the Oracle Intelligent Agent is running on *both* nodes.

Figure 5-3 Console, Management Server and Repository on the Same Machine



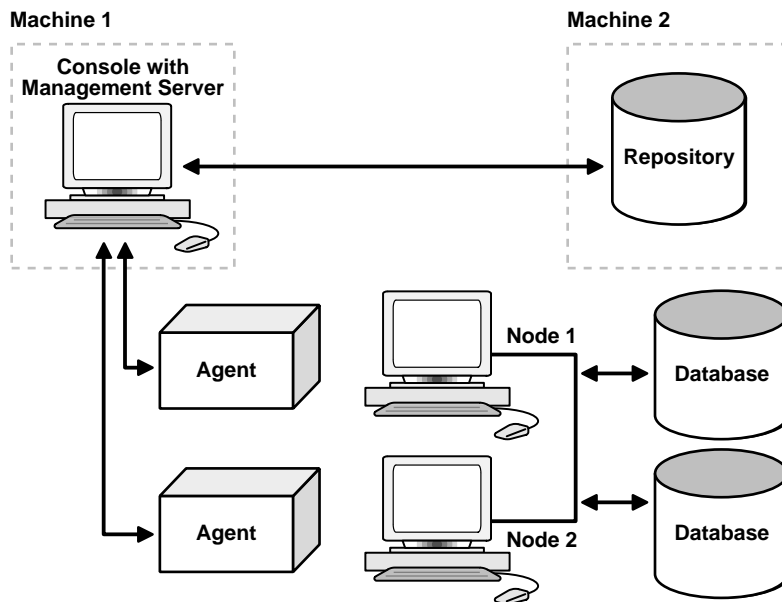
Environment 2: Console, Management Server and repository on separate machines

The Repository is on a separate Windows NT or Solaris machine with an Oracle8i database installed that is only being used as a repository. The Oracle Intelligent Agent does *not* need to be running.

The Management Server remotely manages the databases for the Console on a Windows NT machine. The Console is running on a Windows NT, Windows 95 or Windows 98 machine.

The nodes run an Oracle8 database. The repository is *not* created in this database, and the Oracle Intelligent Agent is running on *both* nodes.

Figure 5-4 Console, Management Server and Repository on the Separate Machines



Installing Oracle Enterprise Manager

To install Oracle Enterprise Manager:

1. Install the repository database on a machine other than one of the Oracle Parallel Server nodes by following the instructions in:
 - Windows NT Server by following the instructions in *Oracle8i Enterprise Edition Installation for Windows NT*
 - Solaris 2.6 or Solaris 2.5.1 by following the instructions in *Oracle8i Installation Guide* for Sun Solaris
2. Install Oracle Enterprise Manager.

You can install the Management Server and the Console on the machine where the repository database resides or individual machines. See *Oracle Enterprise Manager Installation* for complete installation instructions. Oracle Enterprise Manager is a separate CD from the Oracle8i Enterprise Edition CD.

Configuring Oracle Enterprise Manager

To configure Oracle Enterprise Manager, perform these tasks:

- [Step 1: Configure Net8](#)
- [Step 2: Configure OPSCTL for UNIX Platforms](#)
- [Step 3: Configure Repository](#)
- [Step 4: Start Oracle Enterprise Manager Components](#)
- [Step 5: Discover Nodes](#)
- [Step 6: Specify Preferred Credentials for Nodes and Oracle Parallel Server Database](#)

Step 1: Configure Net8

In order for Oracle Parallel Server nodes to be discovered by Oracle Enterprise Manager, a TNSNAMES.ORA file that contains net service name for each instance in the cluster must exist in the `$ORACLE_HOME/network/admin` on UNIX platforms or `ORACLE_HOME\network\admin` on Windows NT. To create this file, follow the procedures in:

["Creating Entries in TNSNAMES.ORA for Each Instance" on page 4-20](#)

["Creating An Entry in the TNSNAMES.ORA File for the Database and Enabling Client Load Balancing" on page 4-25](#)

The Console requires domain-qualified net service names. Therefore, ensure all the net service names are domain qualified.

A sample TNSNAMES.ORA file is shown below:

```
op1.us.oracle.com =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op1))
```

```
op2.us.oracle.com =
(description =
  (address =
    (protocol = tcp)
    (host = opshp2)
    (port = 1521))
(connect_data =
  (service_name = op.us.oracle.com)
  (instance_name = op2))
```

Step 2: Configure OPSCTL for UNIX Platforms

OPCTL requires additional configuration if Oracle Parallel Server is installed on UNIX:

1. Create an `oratab` entry for the name of the Oracle Parallel Server The syntax for this entry is:

```
db_name: oracle_home:N
```

where *db_name* is the database name given to your Oracle Parallel Server database. `oratab` should be in `/etc/oratab` or `/var/opt/oracle/oratab` depending on the platform.

2. You must manually create an `DB_NAME.CONF` file on each node in `$ORACLE_HOME/ops`. Do this for the first node and then FTP this file to all nodes in the cluster.

The OPSCTL utility requires the existence of this file to run properly, even if this file is empty. Depending on your platform, you may need to set the `node_list` parameter. For example,

```
node_list = "1,3-7"
```

Other optional parameters may be set, as described in "[Creating the DB_NAME.CONF File](#)" on page B-2.

3. Ensure that OPSD has been started on each managed node by the `oracle` account at system startup. OPSD resides in `$ORACLE_HOME/bin`.

Step 3: Configure Repository

Use the Configuration Assistant to create and load your version 2 repository. The repository is a set of tables in an Oracle database which stores data required by the Oracle Enterprise Manager. See the *Oracle Enterprise Manager Configuration Guide* for complete configuration instructions.

Step 4: Start Oracle Enterprise Manager Components

To use the Console, start the following components:

- [Oracle Intelligent Agent](#)
- [Management Server](#)
- [Console](#)

Oracle Intelligent Agent

Start the Oracle Intelligent Agent on each of the nodes.

At an operating system prompt, enter the command:

```
lsnrctl dbsnmp_start
```

On Windows NT, you can also use the Control Panel:

1. Double-click the Services icon in the *Control Panel* window.
2. Select the OracleHOME_NAMEAgent service.
3. Click Start to start the service.

Note: Additional Oracle Intelligent Agent configuration is required if the ORACLE_HOME is shared, as described in "[Oracle Intelligent Agent Configuration](#)" on page B-11.

Management Server

Note: Your Management Server service is started automatically during repository creation if you had launched the Configuration Assistant from the Oracle Enterprise Manager installation.

To start the Management Server, at an operating system prompt, enter:

```
oemctrl start oms
```

The command string is case-sensitive and must be entered with lowercase characters.

To start the Management Server in the background on Solaris, enter:

```
% oemctrl start oms&
```

On Windows NT, you can also use the Control Panel:

1. Double-click the Services icon in the *Control Panel* window.
2. Select the Oracle`HOME_NAME`ManagementServer service.
3. Click Start to start the service.

The Management Server looks for the OMSCONFIG.PROPERTIES file in the `$ORACLE_HOME/sysman/config` directory on UNIX and `ORACLE_HOME\sysman\config` directory on Windows platforms. It contains the connect information required by the Management Server to start up.

Note: You should always use the Oracle Enterprise Manager Configuration Assistant to create a repository or to change the repository's connect information. The information is encrypted in the OMSCONFIG.PROPERTIES file. For detailed information editing this file, see the *Oracle Enterprise Manager Configuration Guide*.

Console

The Console gives you a central point of control for the Oracle environment through an intuitive graphical user interface (GUI) that provides powerful and robust system management.

To start the Enterprise Manager Console:

1. At an operating system prompt, enter:

```
oemapp console
```

The command string is case-sensitive and must be entered with lowercase characters.

On Windows NT, you can also choose Start > Programs > Oracle - *HOME_NAME* > Oracle Enterprise Management > Oracle Enterprise Manager Console.

2. If you are logging in to the Console for the first time, after the first login dialog appears, type in the default credentials for the super administrator account:

Administrator = `sysman`

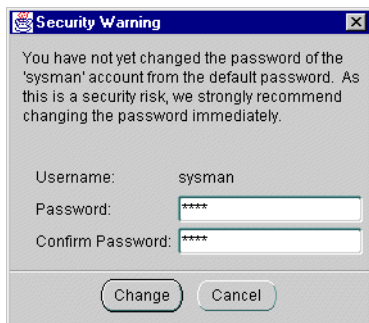
Password = `oem_temp`



3. Select a Management Server which is configured with the repository you want to access from the Management Server drop-down list.

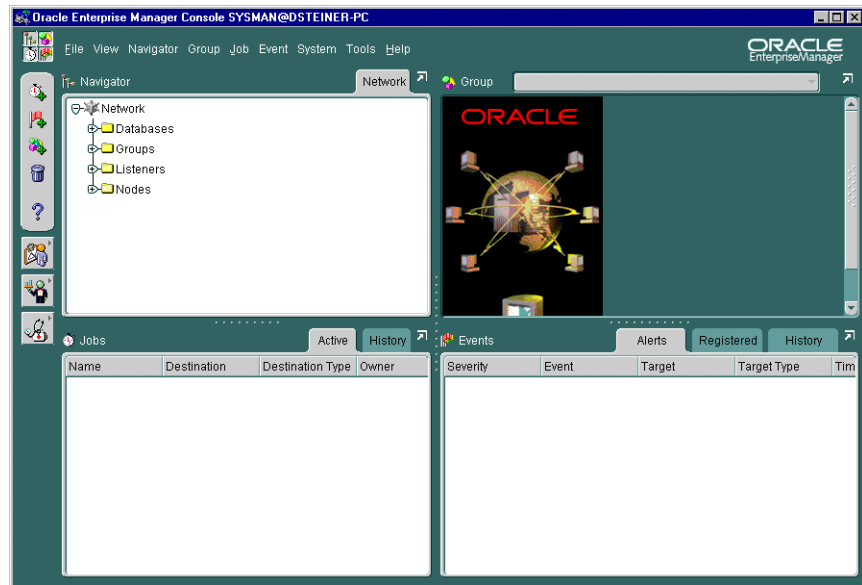
If the name of the Management Server you want to use does not appear, follow the instructions below:

- a. Click the Management Servers button. The Management Servers dialog appears.
 - b. Click the Add button. The Add Management Server dialog appears.
 - c. Type in the name of the Management Server and click OK.
 - d. Select the Management Server from the pull-down list.
4. A *Security Warning* dialog appears where you can change your password. The new password you enter will be the password you use to log in to the Console for subsequent logins.



Note: Passwords must have no more than 8 characters.

The Console appears:



The first time you start the Console, you must login as the super administrator. After other administrator accounts have been created using the super administrator account, you can login as an administrator.

Note: The Console may also be run from a web browser, allowing administrators to run the Console as a "thin" or "fat" client.

Thin clients use a web browser to connect to a server where Console files are installed, whereas fat clients have Console files installed locally. Run from a web browser, the Console enables you to perform all the same administration tasks you would normally perform from the installed Console, including access to the DBA Management Pack applications. You can perform these tasks on any machine that supports a web browser.

For further information about running the Oracle Enterprise Manager from a web browser, following the instructions in the *Oracle Enterprise Manager Configuration Guide*.

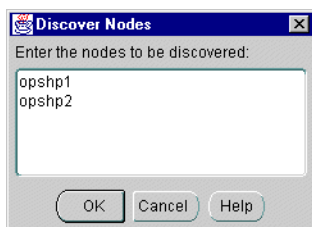
Step 5: Discover Nodes

After the Console has been started, the Console machine must also discover network services, such as databases, listeners, and nodes, to populate the Navigator tree.

To discover services:

1. Choose Discover Node from the Navigator menu.

The *Discover Nodes* dialog appears:

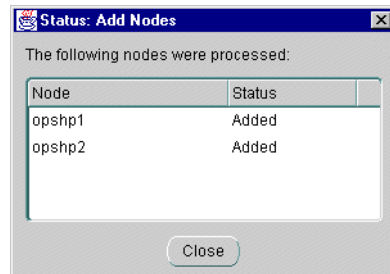


Note: Prior to discovering Oracle Parallel Server nodes, a single-instance must be discovered first. If Databases folder does not exist in the Navigator window, enter the host name of single-instance database. After the single-instance database is discovered, perform discovery of the nodes.

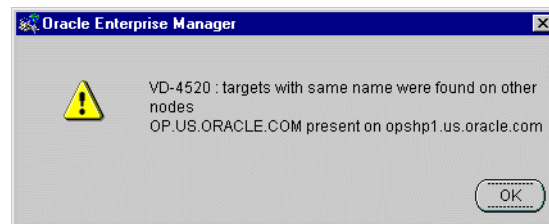
If the repository database is located on the machine with the Console and the Oracle Intelligent Agent is installed and running, the repository database will be discovered automatically.

2. Enter the host names of Oracle Parallel Server nodes.

Successful discovery results in the following:



Ignore the VD-4520 errors for the second and subsequent nodes. The Console considers it an error when a database target is discovered on multiple nodes, but the Oracle Parallel Server discovery reports that the Oracle Parallel Server database exists on each of the nodes. Simply click OK:



An unsuccessful discovery is usually the result of the Oracle Intelligent Agent not being started on the node. For further information about the discovery process and handling errors, see "[Handling Auto-Discovery Failures](#)" on page C-3.

3. Click Close to acknowledge the status dialog.

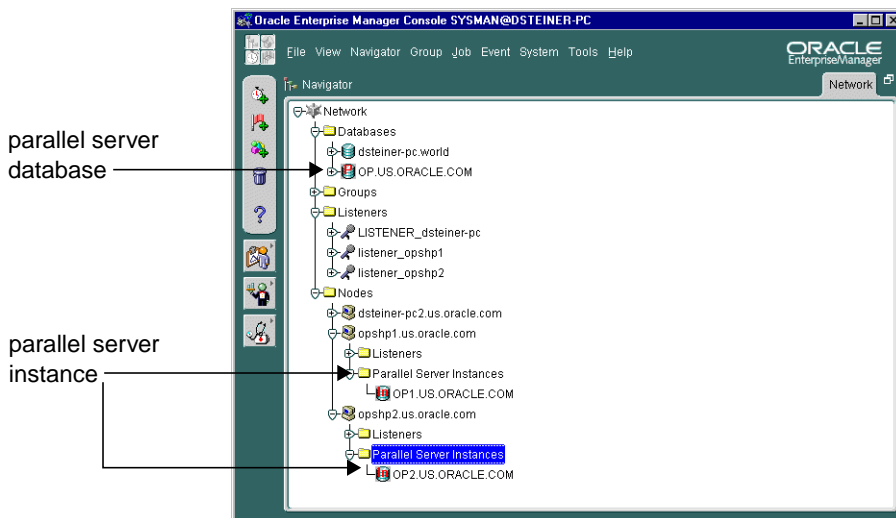
Once a node is identified, Oracle Enterprise Manager automatically discovers (locates) all the databases and other services running on the managed nodes. When a node is discovered, the services located are:

- the database it supports
- its instance
- listener

When an Oracle Parallel Server instance is discovered, a new folder named Parallel Server Instances is created for the node it belongs to.

By expanding and contracting the folders in the Navigator window, you can see the object discovered.

Below is an example of an Oracle Parallel Server database named OP with instance OP1 on host opshp1 and OP2 on opshp2:



Step 6: Specify Preferred Credentials for Nodes and Oracle Parallel Server Database

Oracle Enterprise Manager must be configured with preferred user credentials in order to perform certain functions. These credentials are used when establishing connections to the Oracle Parallel Server, such as when the Oracle Parallel Server is expanded in the Console's navigator. The startup and shutdown operations also use the credentials.

A valid DBA user with SYSDBA or SYSOPER privileges must be identified for the Oracle Parallel Server database in order to expand the Oracle Parallel Server database in the Navigator window or to perform connection and job execution operations. If a valid database user is not identified, you will be prompted each time you try to connect to the database.

Likewise, an operating system user must be identified to run jobs on particular nodes, such as starting or stopping an instance. Although you submit a job from the Console, the job scripts themselves reside on the Oracle Intelligent Agent on the nodes. For this reason, you must have configured a user that has operating system access to the node. You should enter node credentials for each node that was discovered.

This section covers the following topics:

- [Create an Operating System Account](#)
- [Grant SYSDBA or SYSOPER Privileges to a User](#)
- [Set User Credentials in the Console](#)

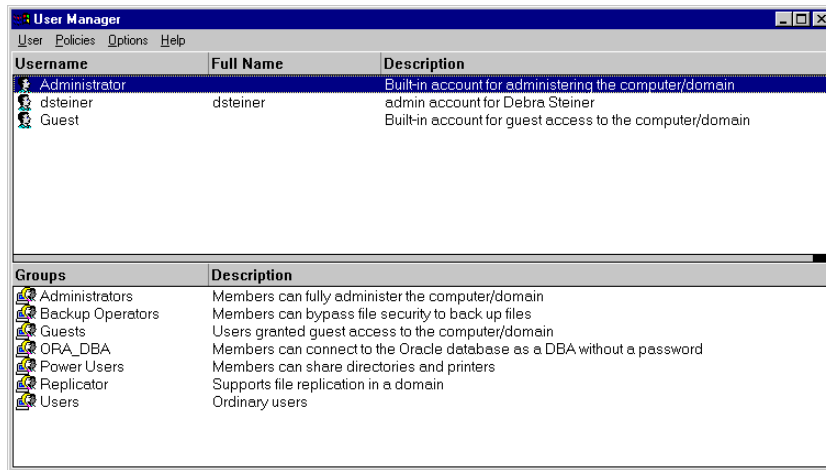
Create an Operating System Account

On UNIX, this user may be the *oracle* account set up during the installation process.

On Windows NT, you will need to create a NT user account in the following fashion. To create a user account on each node of the cluster:

1. Choose Start > Programs > Administrative Tools (Common) > User Manager:

The *User Manager* window appears:



2. Choose New User from the User menu.

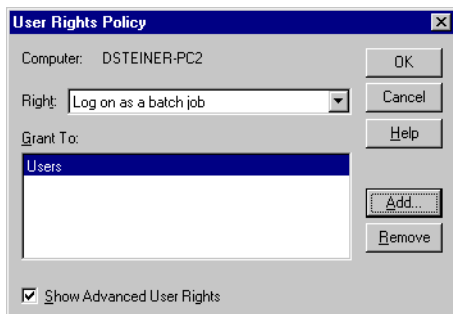
The *New User* dialog box appears:

3. Enter the appropriate information in the dialog box:
 - a. Enter a user name and password.
 - b. Make sure the *User Must Change Password at Next Logon* check box is not checked and *Password Never Expires* check box is checked.
 - c. Make the user a member of the Administrator's group by clicking *Groups*.

The *Group Memberships* dialog box appears:

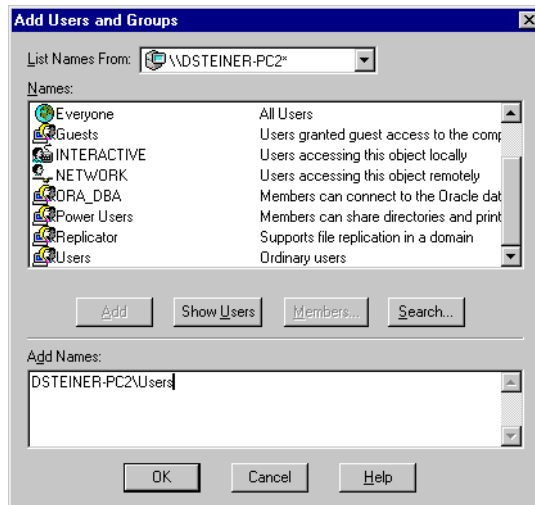
- d. Select *Administrators* from the *Not member of* list box, then click <- Add. *Administrators* is added to the *Member of* list box.

- e. Click OK to close the *Group Memberships* dialog box and return to the *New User* dialog box.
4. Click OK in the *New User* dialog box.
The user is added and the *User Manager* window re-appears.
5. Select the newly-created user, then choose *User Rights* from the Policies menu.
The *User Rights Policy* dialog box appears:



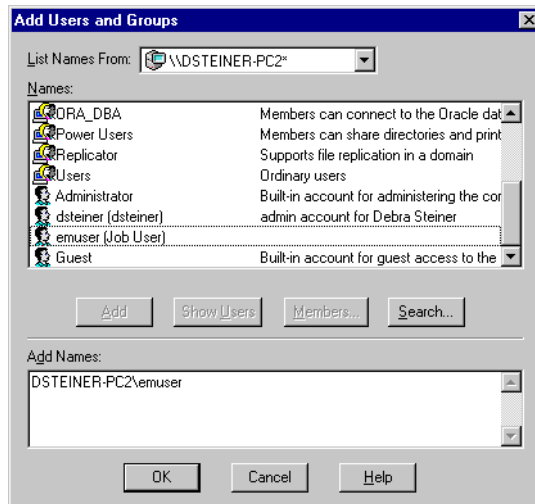
6. Click the *Show Advanced User Rights* check box, then select *Log on as a batch job* from the *Right* drop-down list box.
7. Select the newly created from the *Grant To* list box.
If user does not exist, create it:
 - a. Click *Add*.

The *Add Users and Groups* dialog box appears:



- b. Select the name of the local host machine from the List Names From drop-down list box.
- c. Click the Show Users button.

- d. Select the user from the Names list box, then click Add:



The user displays in the Add Names list box.

- e. Click OK.

The user appears in the Grant To list box in the *User Rights Policy* dialog box.

8. Click OK in the *User Rights Policy* dialog box in the User
The *User Manager* window re-appears.
9. Choose Exit from the User menu.

Grant SYSDBA or SYSOPER Privileges to a User

Identify a current user, such as SYSTEM, or create a new user that will be used to connect, start and stop the database.

Once a user is identified, ensure it has SYSDBA or SYSOPER privileges. SYSDBA and SYSOPER privileges contain all system privileges you will need to manage the database.

To grant SYSDBA or SYSOPER privileges to user, use the SQL GRANT command:

```
SQL> GRANT sysdba to username;
SQL> GRANT sysoper to username;
```

Set User Credentials in the Console

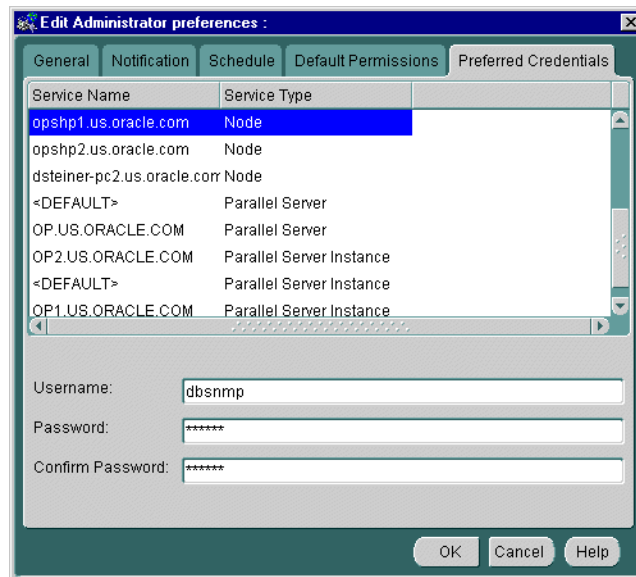
You must set user credentials for the database and each node.

To set credentials:

1. Choose Preferences from the System menu.

The *Edit User Preferences* dialog box appears.

2. Click the Preferred Credentials tab to enter credentials for various service types:



3. Click on a node (identified by the Node service type), and enter the operating system user name and password.
4. Perform step 3 for each node.
5. Click on the Oracle Parallel Server database (identified by the Parallel Server service type).
6. Enter a DBA user name and password that has SYSDBA or SYSOPER privileges for the database, such as SYSTEM/MANAGER, and select SYSDBA or SYSOPER from the Role drop-down menu.

Important: The SYSDBA or SYSOPER privilege is required for Oracle Parallel Server database and instance startup and shutdown.

7. If you plan to use the Oracle Performance Manager application, click on an instance (identify by the Parallel Server Instance service type), enter a user name and password that can connect to instance, select NORMAL role from the drop-down menu.
8. Click OK.

See also:

- ["Managing Instances Using Oracle Parallel Server Management"](#) on page 6-13 for more information about using the Console for Oracle Parallel Server
- *Oracle Enterprise Manager Administrator's Guide* for general Console administration information
- *Oracle Enterprise Manager Configuration Guide* for optional Console configuration

Installing Oracle Performance Manager

To install Oracle Performance Manager:

1. Install the Oracle Diagnostics Pack. See *Oracle Diagnostics Installation* for complete installation instructions. Oracle Diagnostics Pack is on another CD from Oracle Enterprise Manager, requiring a separate license.
2. If you do not plan on using the Oracle Data Gatherer on one of the nodes, install the Oracle Intelligent Agent somewhere on the network. The Oracle Data Gatherer, which is required for Oracle Performance Manager, is automatically installed with the Oracle Intelligent Agent. The Oracle Intelligent Agent may be selected when the Custom install type is chosen in the Oracle8i Enterprise Edition CD-ROM. See the following for further information:
 - *Oracle8i Installation Guide* for Sun Solaris, HP 9000 or AIX-based systems
 - *Oracle8i Enterprise Edition Installation for Windows NT*

Note: The Oracle Performance Manager can be run in addition to or without the Oracle Enterprise Manager. If you choose to run this product stand-alone, Oracle Enterprise Manager does not have to be configured.

Configuring Oracle Performance Manager

Note: Oracle Enterprise Manager and the Oracle Intelligent Agent are *not* required by the Oracle Performance Manager, as its functions are performed using a database connection to the Oracle Parallel Server.

To configure Oracle Performance Manager, perform these tasks:

- [Step 1: Configure Net8](#)
- [Step 2: Start Oracle Performance Manager Components](#)

Step 1: Configure Net8

When Oracle Performance Manager gathers statistics for an Oracle Parallel Server database, it performs a Net8 connection. The machine where Oracle Data Gatherer runs must have a TNSNAMES.ORA file configured with net service names.

A net service name *must* be created for each Oracle Parallel Server instance. To create this file, follow the procedures in "[Creating Entries in TNSNAMES.ORA for Each Instance](#)" on page 4-20.

Step 2: Start Oracle Performance Manager Components

To use the Oracle Performance Manager, start the following components:

- [Oracle Data Gatherer](#)
- [Oracle Performance Manager](#)

Oracle Data Gatherer

Start the Oracle Data Gatherer on at least one of the Oracle Parallel Server nodes.

At an operating system prompt, enter:

```
vppcntl -start
```

On Windows NT, you can also use the Control Panel:

1. Double-click the Services icon in the *Control Panel* window.
2. Select the OracleHOME_NAMEDataGatherer service.
3. Click Start to start the service.

Oracle Performance Manager

1. Start Oracle Performance Manager in one of two ways:

- Standalone

At an operating system prompt, enter:

```
vtm
```

On Windows NT, you can also choose Start > Programs > Oracle - *HOME_NAME* > Oracle Diagnostics Pack > Performance Manager.

The *Performance Manager Login* dialog box appears:



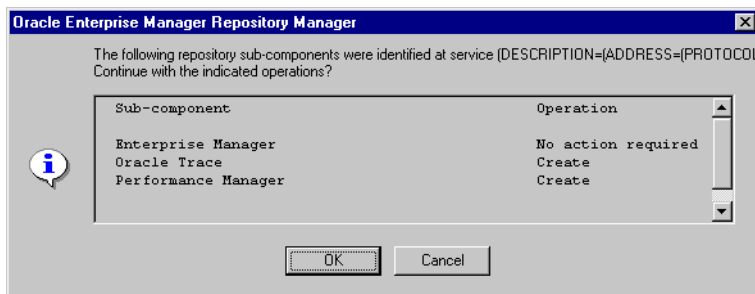
Select whether to log on to the Management Server or in standalone fashion. If *Login to Oracle Management Server* is chosen, additionally enter the Oracle Enterprise Manager administrator user name, password and location of the Management Server.

- Started from Console

Choose Oracle Performance Manager in the Diagnostics Pack toolbar or from the Tools menu choose Diagnostics Pack, and click Oracle Performance Manager.

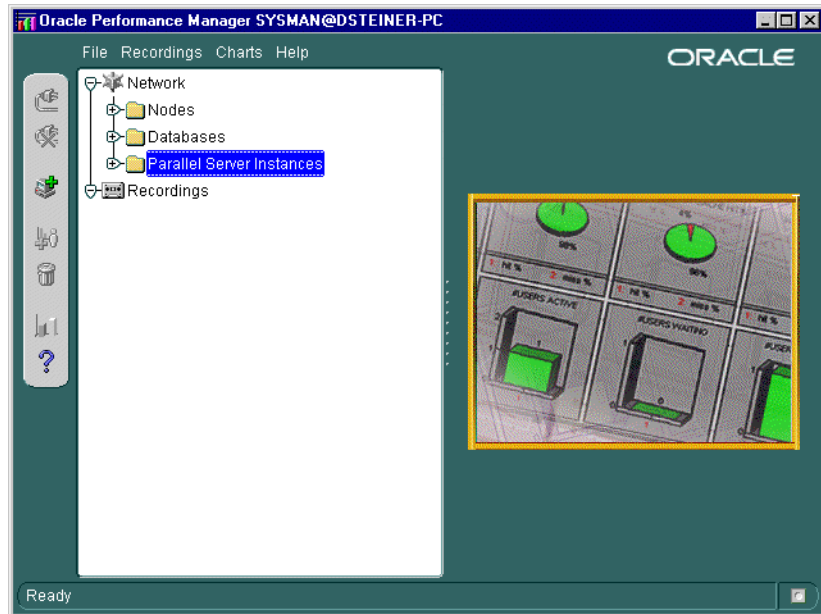
When Oracle Performance Manager is started from the Console, the Oracle Enterprise Manager repository log on credentials are automatically passed to Oracle Performance Manager, thereby providing you with access to all repository-based information. Also, if a service is selected in the Console Navigator window when Oracle Performance Manager is started, Oracle Performance Manager connects to the database.

The first time you log on to Oracle Performance Manager with the *Login to Oracle Management Server* option, the *Oracle Enterprise Manager Repository Manager* window appears, prompting you to build Oracle Performance Manager components in the repository database.



- Click OK to build the Oracle Performance Manager component.

The Oracle Performance Manager starts:



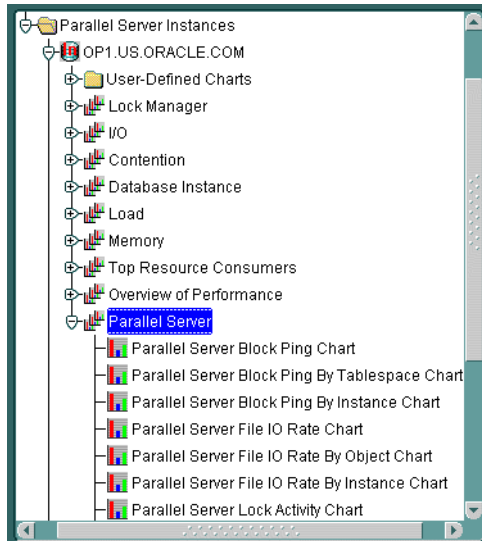
Step 3: Accessing Oracle Parallel Server Charts

Oracle Parallel Server charts may be accessed from the Parallel Server Instance folder or the Databases folder.

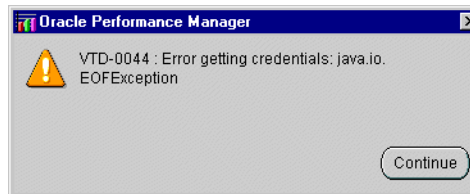
Parallel Server Instances Folder

If the log in occurred from the Console or the *Login to the Oracle Management Server* option was selected in the *Performance Manager Login* dialog box, the Parallel Server Instances folder can be expanded to display the instances, from which the list of available charts may be obtained:

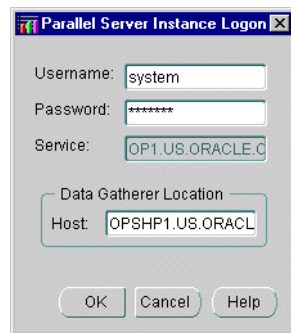
Once the Parallel Server Instances folder is expanded, select User Defined Charts, then select Parallel Server.



If the credentials for the instances was not performed, as described in "[Step 6: Specify Preferred Credentials for Nodes and Oracle Parallel Server Database](#)" on page 5-21, the following error appears when you try to expand an instance:



Click Continue to acknowledge the message. The *Parallel Server Instance Logon* prompts you for the required instance credentials:



Supply the required information, including a database user and password and location of the Oracle Data Gatherer. The Oracle Data Gatherer should be installed and running on some machine in the network. The instance can now be expanded.

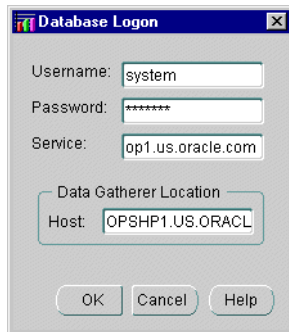
See [Chapter 7, "Monitoring Performance with Oracle Parallel Server Management Charts"](#) for more information about viewing Oracle Parallel Server reports and the *Oracle Enterprise Manager Performance Monitoring and Planning Guide* for more general information about the Oracle Performance Manager application.

Databases Folder

If the log in occurred in standalone mode or the *Standalone, no repository connection* option was selected in the *Performance Manager Login* dialog box, you can access the Oracle Parallel Server charts from the Databases folder:

1. Click the Databases folder.
2. Choose Add New Service from the File menu.

The *Database* dialog prompts you for the required Oracle database credentials:



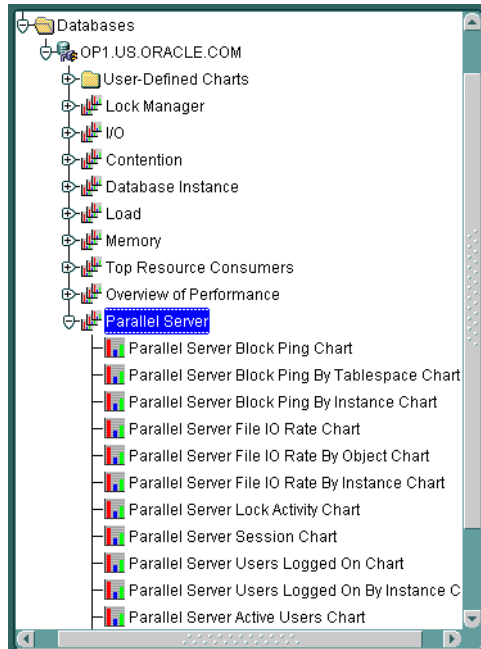
The screenshot shows a 'Database Logon' dialog box with the following fields and values:

- Username: system
- Password: *****
- Service: op1.us.oracle.com
- Data Gatherer Location: Host: OPSHP1.US.ORACL

Buttons at the bottom: OK, Cancel, Help

3. Supply the required information, including a database user and password, net service name in Service field, and location of the Oracle Data Gatherer. The Oracle Data Gatherer should be installed and running on some machine in the network.

4. Expand the Databases folder, select User Defined Charts, then select Parallel Server.



See [Chapter 7, "Monitoring Performance with Oracle Parallel Server Management Charts"](#) for more information about viewing Oracle Parallel Server reports and the *Oracle Enterprise Manager Performance Monitoring and Planning Guide* for more general information about the Oracle Performance Manager application.

Administering Oracle Parallel Server

This chapter describes how instances are managed through initialization files and Oracle Parallel Server Management.

Specific topics covered in this chapter are:

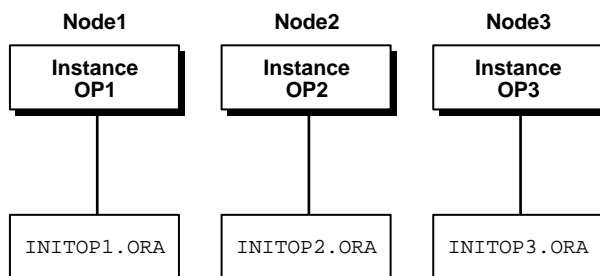
- [Understanding the Initialization Files](#)
- [Setting Initialization Parameters](#)
- [Managing Instances Using Oracle Parallel Server Management](#)

Understanding the Initialization Files

An initialization parameter file is an ASCII text file containing a list of parameters.

Each node consists of an initialization parameter file named `INITSID.ORA` with parameters unique for an instance and an initialization parameter file named `INITDB_NAME.ORA` with common parameters shared from node-to-node.

Figure 6–1 Instance Initialization Files



Purpose of INITSID.ORA

The INITSID.ORA instance initialization parameter files point to the INITDB_NAME.ORA file for common parameters and define the following for each instance:

- unique instance name
- unique thread number
- private rollback segments
- execution of database as an Oracle Parallel Server rather than single instance

The *SID* is the value of the DB_NAME parameter in the INITDB_NAME.ORA file and the thread ID. For instance, if the DB_NAME is OP, and the first node has a thread ID of 1, its SID is OP1; the second node uses the SID OP2 to identify its instance; and so on.

Figure 6–2 and Figure 6–3 show the contents of an initialization files for two nodes with node numbers of 1 and 2 were created by the Oracle Database Configuration Assistant:

Figure 6–2 INITOP1.ORA

```
ifile=C:\Orant3\admin\op\pfile\initop.ora
rollback_segments = (rbs1_1,rbs1_2)
thread=1
parallel_server=true
instance_name=op1
remote_login_passwordfile=exclusive
```

Figure 6–3 INITOP2.ORA

```
ifile=C:\Orant3\admin\op\pfile\initop.ora
rollback_segments = (rbs2_1,rbs2_2)
thread=2
parallel_server=true
instance_name=op2
remote_login_passwordfile=exclusive
```

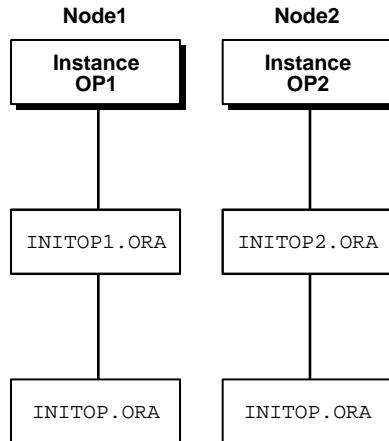
The parameters are described below...

Parameter	Description
IFILE	Identifies the path and name of the <code>INITDB_NAME.ORA</code> to include.
ROLLBACK_SEGMENTS	Specifies one or more rollback segments to allocate to this instance.
THREAD	Specifies the number of the redo thread that is to be used by the instance. Any available redo thread number can be used, but an instance cannot use the same thread number as another instance. Also, an instance cannot start when its redo thread is disabled. A value of zero causes an available, enabled public thread to be chosen. An instance cannot mount a database if the thread is used by another instance or if the thread is disabled.
INSTANCE_NAME	Identifies the name of instance and is used to uniquely identify a specific instance when multiple instances share common services names. <code>INSTANCE_NAME</code> should not be confused with the Oracle System ID (SID), which actually uniquely identifies the instances shared memory on a host.
REMOTE_LOGIN_PASSWORDFILE	Specifies whether Oracle checks for a password file and how many databases can use the password file. This parameter must be set to <code>EXCLUSIVE</code> . <code>EXCLUSIVE</code> specifies that only one instance can use the password file and that the password file contains names other than <code>SYS</code> and <code>INTERNAL</code> . It allows multiple users (other than <code>INTERNAL</code> and <code>SYS</code>) to start up a database.

Purpose of `INITDB_NAME.ORA`

`INITDB_NAME.ORA` database initialization file is called by the individual parameter files through the `IFILE` parameter.

Figure 6–4 Common Initialization Files



For Oracle Parallel Server, some initialization parameters must have the same values for every instance, whether individual or common parameter files are used. By referencing the common parameter file using the `IFILE` parameter within the individual parameter files, instances that have individual parameter files can use the correct parameter values for those that must be identical. This also allows individual parameter files with different values.

Figure 6–5 shows a `INITDB_NAME.ORA` file (`INITOP.ORA`) created for a Hybrid database through the Oracle Database Configuration Assistant:

Figure 6–5 *INITOP.ORA*

```
db_name = op
db_domain = us.oracle.com
service_names = op.us.oracle.com
db_files = 1024
control_files = ("\\.\op_ctl1", "\\.\op_ctl2")
db_file_multiblock_read_count = 8
db_block_buffers = 2843
shared_pool_size = 15728640
log_checkpoint_interval = 10000
log_checkpoint_timeout = 0
processes = 50
parallel_max_servers = 5
log_buffer = 32768
max_dump_file_size = 10240 # limit trace file size to 5M each
global_names = true
oracle_trace_collection_name = ""
background_dump_dest = C:\Orant3\admin\op\bdump
user_dump_dest = C:\Orant3\admin\op\udump

db_block_size = 8192

distributed_transactions = 10
compatible = 8.1.3.0.0
sort_area_size = 66560
```

Parameters to take note of:

Parameter	Description
DB_NAME	Specifies the name of the database, OP, entered during installation or database creation.
DB_DOMAIN	Specifies the database domain, .US.ORACLE.COM, in which the database is located entered during installation or database creation. When possible, Oracle Corporation recommends that your database domain mirror the network domain.
SERVICE_NAMES	<p>Specified the names of the database service on the network, OP.US.ORACLE.COM. By default, the Oracle Universal Installer and the Oracle Database Configuration Assistant create a service name that includes the entire global database name, a name comprised of the database name (DB_NAME) and domain name (DB_DOMAIN), entered during installation or database creation.</p> <p>It is possible to provide multiple services names (by individual SERVICE_NAMES entries) so that different usages of a instance can be identified separately. Service names can also be used to identify a service that is available from multiple instances through the use of replication.</p>

See *Oracle8i Reference* for a description of the other parameters.

Setting Initialization Parameters

This section describes the following:

- GC_* Global Constant Parameters
- Parameter Notes for Multiple Instances
- Identical Parameters on Each Instance
- Parameter Descriptions
- Location of Initialization Files
- Editing Initialization Files

GC_* Global Constant Parameters

Initialization parameters with the prefix *GC* (Global Constant) are relevant only for an Oracle Parallel Server. These parameters are specified in the *INITDB_NAME.ORA* file.

Global constant parameter settings determine the size of the collection of global locks that protect the database buffers on all instances. The settings you choose affect the use of certain operating system resources.

Of the instances (OP1, OP2, and so on), the first instance to start up in shared mode determines the values of the global constant parameters for the other instances. The control file records the values of the *GC_** parameters when the first instance starts up.

When another instance attempts to start up in shared mode, the Oracle Server compares the values of the global constant parameters in its parameter file with those already in use and issues a message if any values are incompatible. The instance cannot mount the database unless it has the correct values for its global constant parameters.

The global constant parameters for an Oracle Parallel Server are:

Parameter	Description
GC_FILES_TO_LOCKS	Gives the mapping of hashed and fine-grain locks to blocks within each data file. The meaning of this parameter has changed. Previously, files not mentioned in this parameter (or files added later) were assigned the remaining hash locks. Files not mentioned in this parameter use DBA locking. You can now have multiple entries of GC_FILES_TO_LOCKS.
GC_RELEASABLE_LOCKS	Sets the number of locks which will be used for DBA locks.
GC_ROLLBACK_LOCKS	For each rollback segment, specifies the number of instance locks available for simultaneously modified rollback segment blocks.

Parameter Notes for Multiple Instances

Multiple instance issues concerning initialization parameters are summarized in the following table:

Parameter	Parallel Server Notes
CHECKPOINT_PROCESS	In Oracle Parallel Server, your database can have more datafiles. To speed up checkpoints, enable the CHECKPOINT_PROCESS parameter.
DELAYED_LOGGING_BLOCK_CLEANOUTS	If set to TRUE, this parameter can potentially reduce pinging between instances.
DML_LOCKS	This parameter must be identical on all instances only if set to zero.
INSTANCE_NUMBER	If specified, this parameter must have unique values for different instances.
LOG_ARCHIVE_FORMAT	You must include the thread number.
MAX_COMMIT_PROPAGATION_DELAY	If you want commits to be seen immediately on remote instances, you may need to change the value of this parameter.
NLS_* parameters	This parameter can have different values for different instances.
PARALLEL_SERVER	To enable a database to be started in Oracle Parallel Server mode, this parameter must be set to TRUE in the initialization file.
PROCESSES	This parameter must have a value large enough to allow for all background processes and all user processes in an instance. Some operating systems can have additional DBWR processes. Defaults for the SESSIONS and TRANSACTIONS parameters are derived directly or indirectly from the value of the PROCESSES parameter. If you do not use the defaults, you may want to increase some of these parameter values to allow for optional background processes.
RECOVERY_PARALLELISM	To speed up the roll forward or cache recovery phase, you can set this parameter.
ROLLBACK_SEGMENTS	Specify the private rollback segments for each instance.
THREAD	If specified, this parameter must have unique values for different instances.

Identical Parameters on Each Instance

Certain initialization parameters are critical at database creation or affect certain database operations. These parameters must have the same value for every instance in an Oracle Parallel Server specified in each `INITDB_NAME.ORA` file for each instance. For example, the values of `DB_BLOCK_SIZE` and `CONTROL_FILES` must be identical for every instance.

The following initialization parameters must have identical values (unless noted otherwise) for every instance in an Oracle Parallel Server.

<code>CONTROL_FILES</code>	<code>LM_LOCKS</code> (identical values recommended)
<code>CPU_COUNT</code>	<code>LM_PROCS</code> (identical values recommended)
<code>DB_BLOCK_SIZE</code>	<code>LM_RESS</code> (identical values recommended)
<code>DB_FILES</code>	<code>MAX_COMMIT_PROPAGATION_DELAY</code>
<code>DB_NAME</code>	<code>PARALLEL_DEFAULT_MAX_INSTANCES</code>
<code>DB_DOMAIN</code>	<code>PARALLEL_DEFAULT_MAX_SCANS</code>
<code>DML_LOCKS</code>	<code>ROLLBACK_SEGMENTS</code>
<code>GC_FILES_TO_LOCKS</code>	<code>ROW_LOCKING</code>
<code>GC_ROLLBACK_LOCKS</code>	

Setting LM_* Parameters

Set values for the LM_* initialization parameters. Note that the resources, locks and processes configurations are per Oracle Parallel Server instance. For ease of administration, these parameters should be consistent for all the instances.

Parameter	Description
LM_LOCKS	Number of locks. Where R is the number of resources, N is the total number of nodes, and L is the total number of locks, the following calculation is used: $L = R + (R*(N - 1))/N$
LM_PROCS	Number of processes. The value of PROCESSES initialization parameter multiplied by the number of nodes.
LM_RESS	This parameter controls the number of resources that can be locked by the Lock Manager. This parameter covers the number of lock resources allocated for DML, DDL (data dictionary locks), and data dictionary cache locks + file and log management locks.

Parameter Descriptions

Chapter 18, "Administering Multiple Instances," of the *Oracle8i Parallel Server Concepts & Administration* includes descriptions of the initialization parameters.

Location of Initialization Files

The database for which the instance is started must have access to the appropriate initialization parameter files. Oracle Parallel Server uses the initialization parameter files located in:

- `$ORACLE_BASE/admin/db_name/pfile` on UNIX
- `ORACLE_BASE\admin\db_name\pfile` on Windows NT

Editing Initialization Files

To customize Oracle Parallel Server databases functions, you need to edit the initialization parameter files. Use any ASCII text editor to modify the files.

Managing Instances Using Oracle Parallel Server Management

Oracle Parallel Server Management allows you to perform a variety of management tasks on your Oracle Parallel Servers, distributed systems, and databases. The Enterprise Manager Console provides a central point of control for the Oracle environment through an intuitive graphical user interface (GUI) that provides drag-and-drop system management.

The Console enables you to manage a heterogeneous environment as easily as a homogeneous one. You can schedule jobs on multiple nodes simultaneously or monitor groups of services together.

A database or system administrator can thus handle all nodes in an Oracle Parallel Server as a single entity. For example, you can configure Oracle Enterprise Manager to execute a job across all nodes of an Oracle Parallel Server.

This section is meant to only describe Oracle Enterprise Manager administration for Oracle Parallel Server. Use this section as a supplement to general information contained in the *Oracle Enterprise Manager Administrator's Guide*. This section covers the following topics:

- [Starting the Console](#)
- [Displaying Oracle Parallel Server Objects in the Navigator Window](#)
- [Starting an Oracle Parallel Server Database](#)
- [Shutting Down an Oracle Parallel Server Database](#)
- [Viewing the Parallel Server Operation Results](#)
- [Viewing Oracle Parallel Server Status](#)
- [Creating a Job for a Parallel Server or Parallel Server Instance](#)
- [Specifying Job Details](#)

See [Chapter 5, "Installing and Configuring Oracle Parallel Server Management"](#), for configuration information.

Starting the Console

To use the Console, start the following components:

- Oracle Intelligent Agents on each of the nodes
- Management Server
- Console

See "[Step 4: Start Oracle Enterprise Manager Components](#)" on page 5-13 for instructions.

Displaying Oracle Parallel Server Objects in the Navigator Window

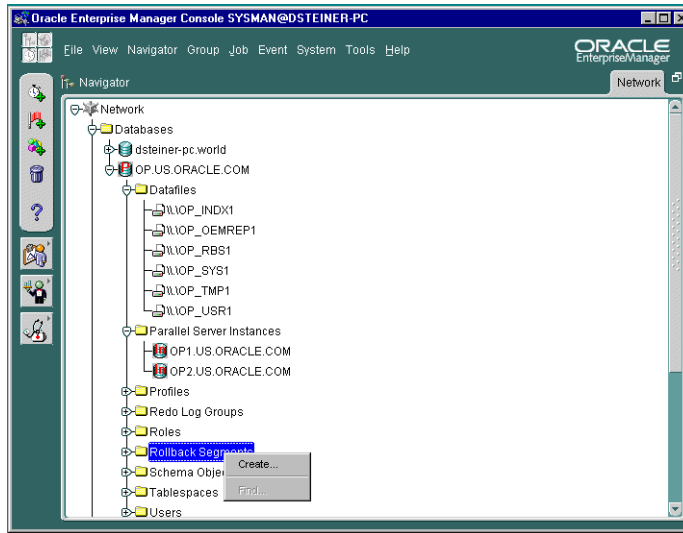
From the Navigator window, you can view and manage both single-instance and multiple-instance databases. The information available for Oracle Parallel Servers is the same as for single-instance databases.

The Navigator displays all the network objects and their relationships to other objects including a direct view of objects such as user-defined groups, nodes, listeners, servers, databases, and database objects.

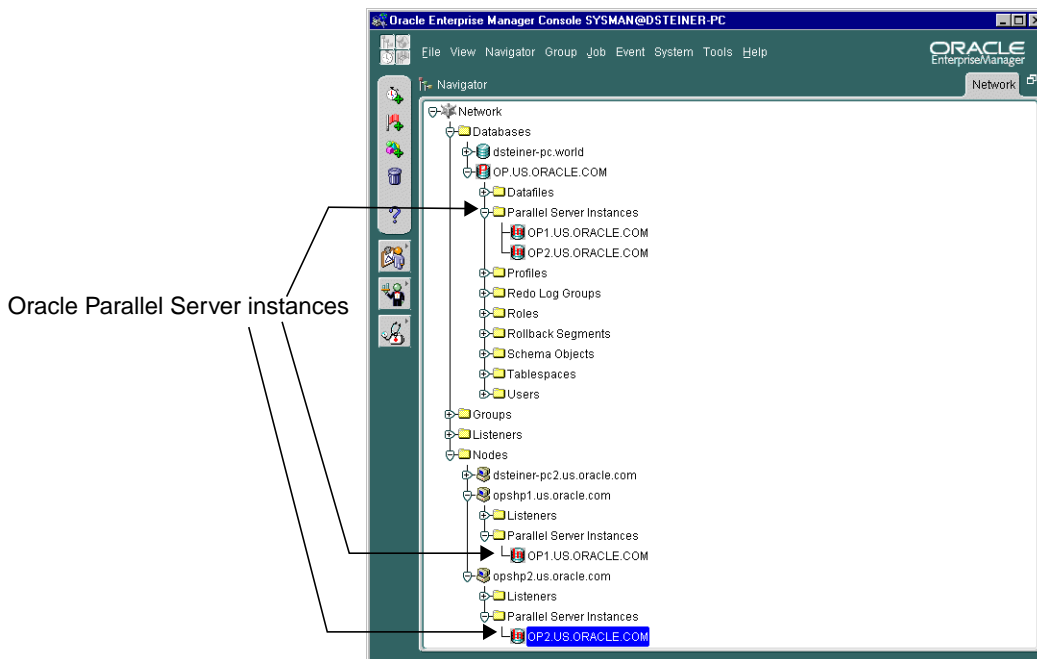
From the Navigator window, Oracle Parallel Servers are located in the Databases folder which contains entries for single-instance and Oracle Parallel Server databases. Each Oracle Parallel Server folder contains the objects for data files, in-doubt transactions, profiles, redo log groups, roles, rollback segments, schema objects, tablespaces, and users.

Note: If preferred credentials were not set for the database, as described in "[Step 6: Specify Preferred Credentials for Nodes and Oracle Parallel Server Database](#)" on page 5-21, the *Database Connect Information* dialog prompts you to enter database connect information.

An Oracle Parallel Server database's subfolders, such as Datafiles and Users, behave just as they do for single-instance databases. By right-clicking the mouse on these objects, property sheets can be accessed to inspect and modify properties of these object just as for single-instance database. For example, right-clicking on the Redo Log Groups folder and choosing Create to add a new redo log group

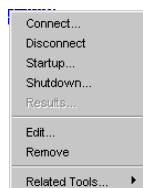


The Oracle Parallel Server database and each discovered Oracle Parallel Server node contains a Parallel Server Instances folder that can be expanded to display the instances belonging to the database or node.



Using the Right-Mouse Menu

Right-clicking on a Oracle Parallel Server brings up the Oracle Parallel Server right-mouse menu:



This menu contains entries for the following functions:

Option	Description
Connect	Connects to the database. In the <i>Database Connect Information</i> dialog box, enter the database user name, password and privileges. If you want to perform a startup or shutdown, a user with SYSDBA privileges must be specified.
Disconnect	Terminates a connection to the database.
Startup	Starts the database. See " Starting an Oracle Parallel Server Database " on page 6-18 for Oracle Parallel Server-related instructions.
Shutdown	Shuts down the database. See " Shutting Down an Oracle Parallel Server Database " on page 6-21 for Oracle Parallel Server-related instructions.
Results	Displays startup and shutdown results, as described in " Viewing the Parallel Server Operation Results " on page 6-24.
Edit	Allows inspection of the state of the Oracle Parallel Server, including which instances are active. See " Viewing Oracle Parallel Server Status " on page 6-27 for Oracle Parallel Server-related instructions.
Remove	Deletes the database object and its related services. This should only be performed if it is no longer necessary to monitor or manage a database from Oracle Enterprise Manager.
Related Tools	Contains access to other tools which have been enabled for Oracle Parallel Server.

Starting an Oracle Parallel Server Database

The Console allows you to start an Oracle Parallel Server database or individual instances.

Once all Oracle Parallel Server instances are up, the Oracle Parallel Server database is considered to be up.

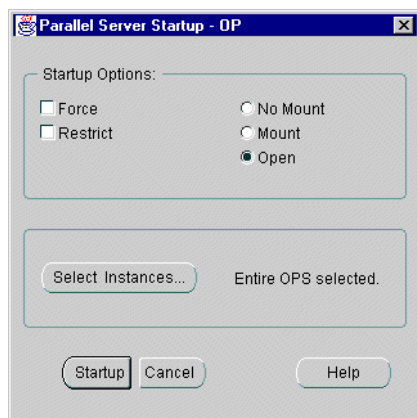
To start up an Oracle Parallel Server database:

1. Right-click on an Oracle Parallel Server database under the Databases folder in the Navigator window.

A menu appears with options for the database.

2. Choose Startup from the menu.

The *Parallel Server Startup* dialog box appears.



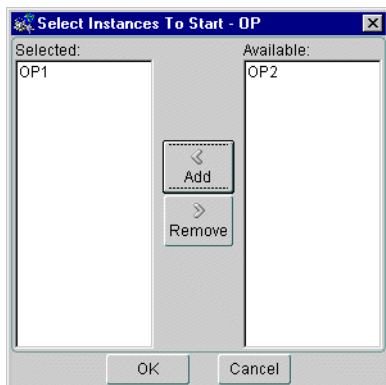
3. Select the startup type:

Option	Description
Force	Shuts down the currently running Oracle instances with the SHUTDOWN mode, ABORT, before restarting them. If the instances are running and FORCE is not specified, an error results. Warning: You should <i>not</i> use the FORCE mode under normal circumstances. Use the FORCE mode only while debugging and under abnormal circumstances.
Restrict	Makes the started instances accessible only to users with the RESTRICTED SESSION system privilege. Users already connected are not affected.
No Mount	Does not mount the database upon instance startup.
Mount	Mounts a database but does not open it.
Open	(<i>default</i>) Opens the database.

4. If you want to start up all instances, click Startup. If you want to start up only selected instances, follow these steps:

a. Select Instances.

The *Select Instances to Start* dialog box appears:



b. Select the instances to startup in the Available list, then click Add.

- c. Click OK to close the *Select Instances to Start* dialog box.
- d. Click Startup from the *Parallel Server Startup* dialog box.

The *Parallel Server Operation Results* dialog box displays the progress of the startup operation, as described in "[Viewing the Parallel Server Operation Results](#)" on page 6-24.

If the instances were started successfully, the *Parallel Server Started* message dialog box appears with a successful message:



Click OK in the *Parallel Server Started* message to acknowledge the message, then click Close in the *Parallel Server Operation Results*.

If the startup fails, the *Parallel Server Started* message box appears with a failure message. Click View Details to view more information in the *Parallel Server Operation Results* dialog about why the shutdown failed, then click Close.

Shutting Down an Oracle Parallel Server Database

The Console allows you to shut down an Oracle Parallel Server database or individual instances.

Once all Oracle Parallel Server instances are shut down, the Oracle Parallel Server is considered to be shut down.

Note: Occasionally, an Oracle Parallel Server database may be completely down, but some of its services, such as the database listener, may remain running.

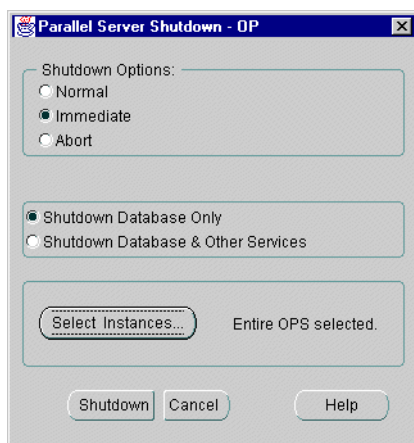
To shut down an Oracle Parallel Server database:

1. Right-click on an Oracle Parallel Server database under the Databases folder in the Navigator window.

A menu appears with options for the database.

2. Choose Shutdown from the menu.

The *Parallel Server Shutdown* dialog box appears.



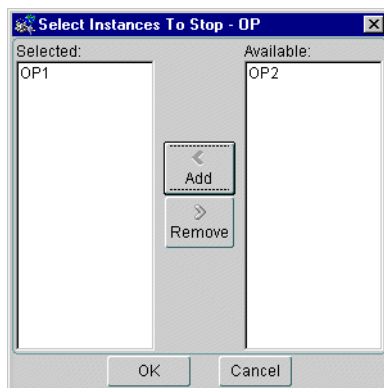
3. Select the shutdown type:

Option	Description
Normal	Waits for the currently connected users to disconnect from the database, prohibits further connects, and closes and dismounts the database before shutting down the instance. Instance recovery is not required on next startup.
Immediate	<i>(default)</i> Does not wait for current calls to complete, prohibits further connects, and closes and dismounts the database. The instance is immediately shut down. Connected users are not required to disconnect and instance recovery is not required on next startup.
Abort	Proceeds with the fastest possible shut down. Connected users are not required to disconnect. The database is not closed or dismounted, but the instances are shut down. Instance recovery is required on next startup. Warning: You must use this option if a background process terminates abnormally.
Shutdown Database Only	<i>(default)</i> Shuts down the database only. The services required for an instance, such as the listener remain up and available.
Shutdown Database & Other Services	Shuts down the database and associated services, such as the listener.

4. If you want to shut down all instances, click Shutdown. If you want to start up only selected instances, follow these additional steps:

- a. Select Instances.

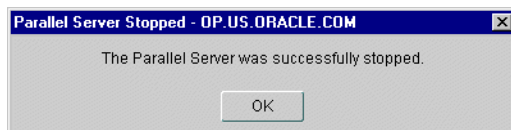
The *Select Instances to Stop* dialog box appears:



- b. Select the instances to stop in the Available list, then click Add.
- c. Click OK to close the *Select Instances to Stop* dialog box.
- d. Click Shutdown from the *Parallel Server Shutdown* dialog box.

The *Parallel Server Operation Results* dialog box displays the progress of the shutdown operation, as described in "[Viewing the Parallel Server Operation Results](#)" on page 6-24.

If the instances were shut down successfully, the *Parallel Server Stopped* message dialog box appears with a successful message.:



Click OK in the *Parallel Server Stopped* message to acknowledge the message, then click Close in the *Parallel Server Operation Results*.

If the shutdown fails, the *Parallel Server Stopped* message box appears with a failure message. Click *View Details* to view more information in the *Parallel Server Operation Results* dialog about why the shutdown failed, then click *Close*

Viewing the Parallel Server Operation Results

The *Parallel Server Operation Results* dialog displays information about the progress of the instance startup or shutdown operation you selected:

The operation results are presented in two views:

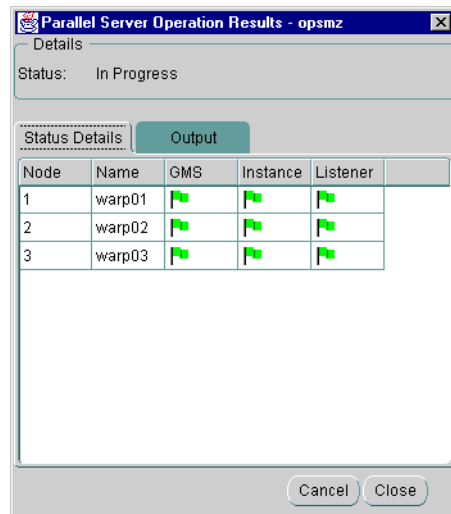
- [Status Details Tab](#)
- [Output Tab](#)

The *Parallel Server Operation Results* dialog box automatically displays during a startup or shutdown operation. It may also be initiated with the following steps:

1. Right-click on an Oracle Parallel Server database under the *Databases* folder in the Navigator window.
A menu appears with options for the database.
2. Choose *Results* from the menu.

Status Details Tab

While the startup operation is running against an Oracle Parallel Server running on a UNIX system, the Status Details tab progress display is shown, with the status of object startup, and updated dynamically as the operation progresses:



The services managed by Oracle Parallel Server Management vary by platform and by version. The Oracle Parallel Server being managed in this example is release 8.0.4, so the GMS (Group Membership Service) is being started. In release 8.1, the GMS is built into the RDBMS kernel, so only instance and listener columns display.

When shutting down a Oracle Parallel Server, the tab displays only those nodes and instances that are being affected by the shutdown operation.

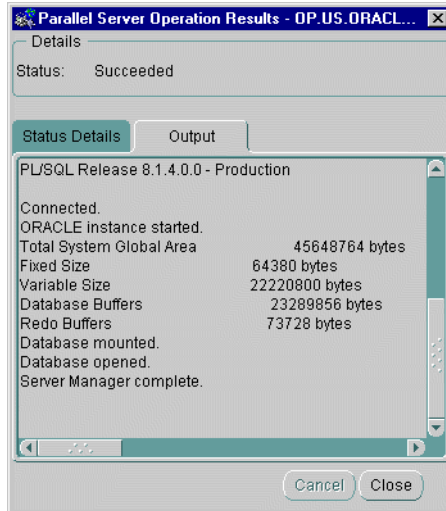
Note: This tab is not available for Windows NT Oracle Parallel Servers, because OPSCCTL on Windows NT does not generate status details.

The following are the possible states that each component may experience:

State	Description
Up (green flag)	The component is running.
Down (red flag)	The component is not running.
Can't determine (gray flag)	Oracle Enterprise Manager cannot determine the state of the component. This state occurs typically when the component startup or shutdown operation has not completed.
Component does not exist on this node (blank background)	The component was not configured on the node. Not all components (listener, instance) are required to exist on every node.

Output Tab

The Output tab displays the commands executed by the Oracle Parallel Server node and any associated error messages in textual format.



Viewing Oracle Parallel Server Status

The *Edit Parallel Server* dialog box displays status information about the parallel database, such as instances available in the Oracle Parallel Server and status of Oracle Parallel Server components.

Note: Because this dialog box requires a connection to an Oracle Parallel Server, this dialog box will not appear if the Oracle Parallel Server is down.

To view status information about an Oracle Parallel Server database:

1. Right-click on an Oracle Parallel Server database under the Databases folder in the Navigator window.

A menu appears with options for the database.

2. Choose Edit from the menu.

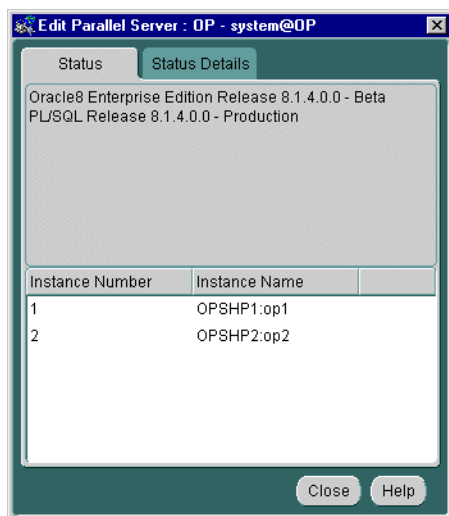
The *Edit Parallel Server* dialog box appears.

The operation results are presented in two views:

- [Status Tab](#)
- [Status Details Tab](#)

Status Tab

The Status tab displays information about the currently running instances by querying V\$ACTIVE_INSTANCES table. Oracle Enterprise Manager makes a connection to the Oracle Parallel Server, therefore, this tab will not appear if the Oracle Parallel Server is down.

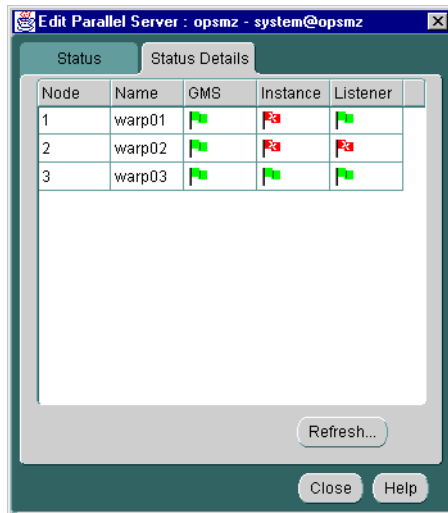


Column Name	Description
Instance Number	The number assigned to the instance.
Instance Name	The name specified for the instance and the node it is running on. This name has the following format: <i>node:instance_name</i> .

Status Details Tab

Note: This tab is not available for a Windows NT Oracle Parallel Server, because OPSCCTL on Windows NT does not generate status details.

Displays an overall view of the state of the Oracle Parallel Server and related components. This tab displays the status of the various components, such as listeners, and instances for all nodes.



The following are the possible states that each component may experience:

State	Description
Up (green flag)	The component is running.
Down (red flag)	The component is not running.
Can't determine (gray flag)	Oracle Enterprise Manager cannot determine the state of the component. This state occurs typically when the component startup or shutdown operation has not completed.
Component does not exist on this node (blank background)	The component was not configured on the node. Not all components (listener, instance) are required to exist on every node.

Creating a Job for a Parallel Server or Parallel Server Instance

The job scheduling system provides a highly reliable and flexible mechanism for DBAs to schedule and automate repetitive jobs on both the Oracle Parallel Server database and Oracle Parallel Server instances.

The Console contains a full-featured scheduling tool which allows DBAs to come up with a customized schedule. This provides DBAs with true “lights out” management capability so that they can focus on other tasks. A rich selection of jobs is provided for Oracle Parallel Servers.

You can create a job with a Oracle Parallel Server database or an Oracle Parallel Server instance as the destination. To create a new job, follow these steps:

1. Choose Create Job from the Job menu.
2. Complete the tabs of the Create Job property sheet.
3. When you are satisfied with your job settings, click the Submit button to submit the job to the Oracle Intelligent Agent. The job appears in the Active Jobs window.
4. Click the Save button to save the job. The job appears in the Job Library window. You can modify or submit a saved job at a later time.

Note: There is usually a slight delay between submitting the job and notification by the Oracle Intelligent Agent.

Specifying Job Details

From the Create Job property sheet, you can specify the details of a new job. The Create Job property sheet contains these tabs:

Tab	Description
General	Specify the job name, description, destination type, and destination.
Tasks	Choose the task(s) that you want the job to perform.
Parameters	Set the run-time parameters for the tasks. The parameters that appear on this tab depend on which task(s) you chose on the Task list box.
Schedule	Schedule the time and frequency you want Oracle Enterprise Manager to run the job.
Permissions	Specify the administrator to perform the job.

The following tabs contain Oracle Parallel Server-specific options:

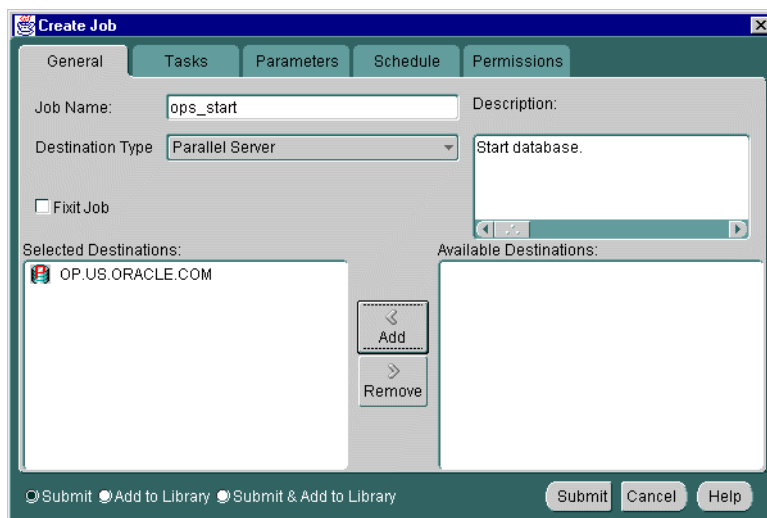
- [General Tab](#)
- [Tasks Tab](#)
- [Parameters Tab](#)

See the *Oracle Enterprise Manager Administrator's Guide* for general job scheduling information.

General Tab

From the General tab, specify the:

- Job Name
- Destination Type (either Parallel Server or Parallel Server Instance)
- Description
- Destinations



The screenshot shows the 'Create Job' dialog box with the 'General' tab selected. The 'Job Name' field contains 'ops_start'. The 'Destination Type' dropdown is set to 'Parallel Server'. The 'Description' field contains 'Start database.'. The 'Selected Destinations' list contains 'OP.US.ORACLE.COM'. The 'Available Destinations' list is empty. The 'Add' and 'Remove' buttons are visible between the two destination lists. At the bottom, there are radio buttons for 'Submit', 'Add to Library', and 'Submit & Add to Library', along with 'Submit', 'Cancel', and 'Help' buttons.

General | Tasks | Parameters | Schedule | Permissions

Job Name: ops_start Description: Start database.

Destination Type: Parallel Server

Fixit Job

Selected Destinations: OP.US.ORACLE.COM

Available Destinations:

Add Remove

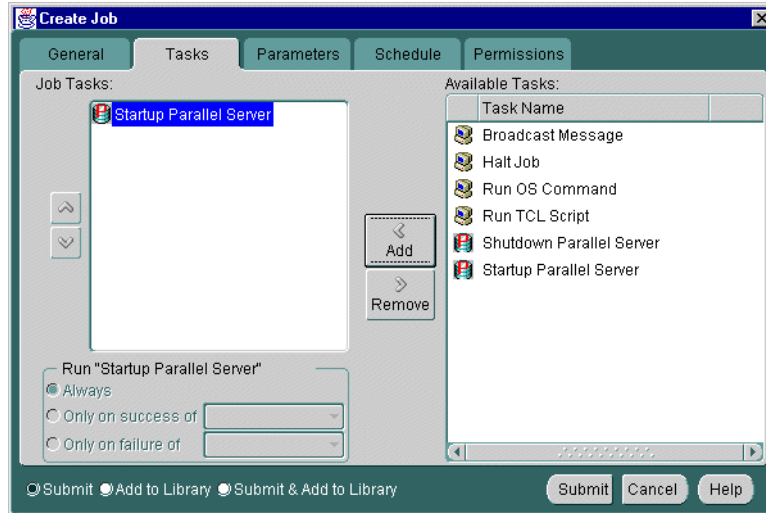
Submit Add to Library Submit & Add to Library Submit Cancel Help

The General tab contains these options:

Parameter	Description
Job Name	Enter the name of the new job.
Description	Enter a description of the job.
Destination Type	Select a destination type from the drop-down list box. You can choose from the following options: parallel server, parallel server instance, database, listener, or node.
Available Destinations	<p>The destinations are determined by your selection of the Destination Type. The destinations include parallel servers, parallel server instances, databases, listeners, and nodes.</p> <p>Click the destinations of the job in the Available Destinations list, then click Add to move the destination to the Selected Destinations list. To remove a destination from a job, click the destination in the Selected Destinations list, then click Remove.</p>
Fixit Job	Select this check box if you want to use this job as the fixit job for an event occurrence. The job cannot be scheduled.

Tasks Tab

From the Tasks tab, choose the task(s) that you want the job to perform. The list of tasks that appear is different depending on whether you select an Oracle Parallel Server or an Oracle Parallel Server instance as your Destination Type from the General tab.



Move the tasks between the Available Tasks and Selected Tasks lists with the Add and Remove buttons.

Tasks for Parallel Server Destinations If your Destination Type is a Parallel Server, you can choose from these tasks:

- Broadcast Message
- Halt Job
- Run OS Command
- Run TCL Script
- Shutdown Parallel Server
- Startup Parallel Server

Tasks for Parallel Server Instance Destinations If your Destination Type is a Parallel Server Instance, you can choose from these tasks:

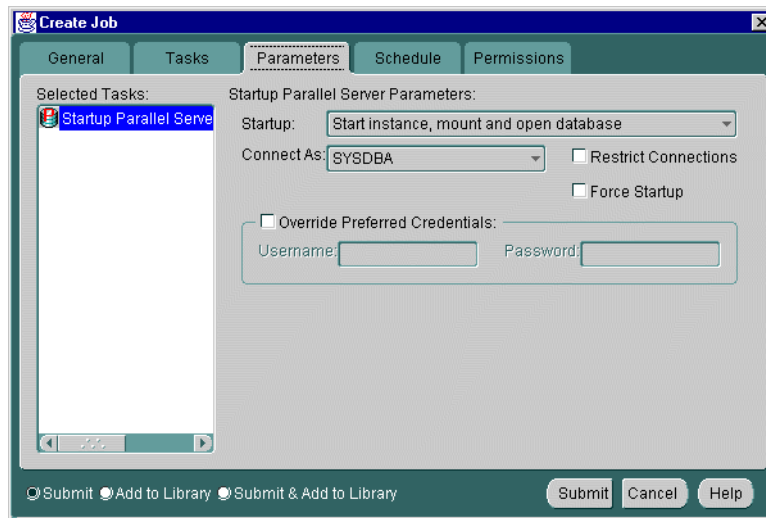
- Broadcast Message
- Halt Job
- Run OS Command
- Run TCL Script
- Shutdown Parallel Server
- Startup Parallel Server

For a description of these tasks and the parameters you need to set, see the *Oracle Enterprise Manager Administrator's Guide*.

Parameters Tab

From the Parameters tab, you can specify parameter settings for the job tasks you selected on the Tasks tab. The parameters which display vary according to the job task. Parameters for Oracle Parallel Server startup and shutdown tasks are described below. To find out what parameters you need to set for Oracle Parallel Server instance tasks, refer to the *Oracle Enterprise Manager Administrator's Guide*.

Parallel Server Startup Task When you select the Startup Parallel Server task on the Tasks tab, the following displays:



The screenshot shows the 'Create Job' dialog box with the 'Parameters' tab selected. The 'Selected Tasks' list on the left contains 'Startup Parallel Serve'. The 'Startup Parallel Server Parameters' section on the right includes the following fields and options:

- Startup:** A dropdown menu set to 'Start instance, mount and open database'.
- Connect As:** A dropdown menu set to 'SYSDBA'.
- Restrict Connections**
- Force Startup**
- Override Preferred Credentials:** A sub-section containing:
 - Username:** An empty text input field.
 - Password:** An empty text input field.

At the bottom of the dialog, there are three radio buttons: 'Submit', 'Add to Library', and 'Submit & Add to Library'. To the right of these are three buttons: 'Submit', 'Cancel', and 'Help'.

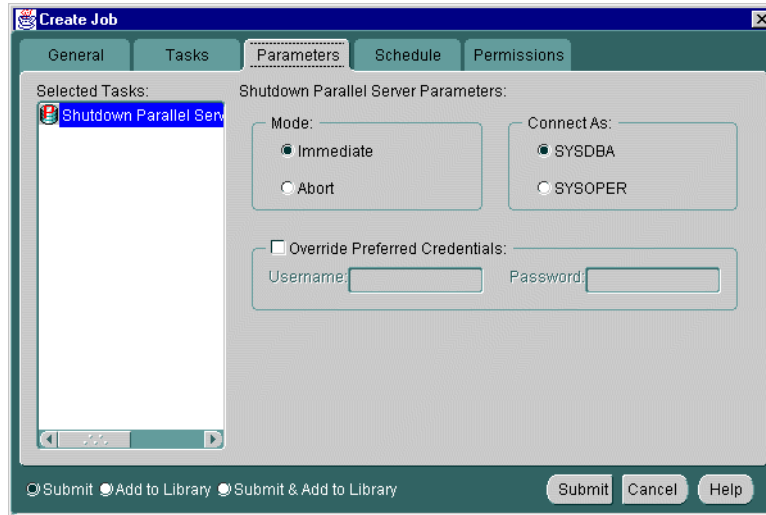
Complete the parameters on the tab and click the Submit button to run the Oracle Parallel Server startup task.

The Parameters tab for Startup contains these options:

Parameters	Description
Startup	Select the startup modes for the job from the drop-down list box.
Connect As	<p data-bbox="586 371 753 399">Specify the role.</p> <p data-bbox="586 413 1268 569">Only Normal is allowed for Oracle7. For Oracle8i, SYSOPER and SYSDBA roles allow you the maximum database administration privileges. You require SYSDBA or SYSOPER privileges to run job tasks such as shutdown or startup on the database. For more information about SYSOPER and SYSDBA roles, refer to the <i>Oracle8i Server Administrator's Guide</i>.</p> <p data-bbox="586 586 1268 690">If you try to connect as SYSDBA and do not have SYSDBA privileges, an error message states that an invalid user name or password was entered. The error message should state that invalid login credentials were entered.</p>
Override Preferred Credentials	You can use the preferred credentials that have been set up for the database, or you can enter another database user name and password.

Parameters for the Parallel Server Shutdown Task

When you select the Shutdown Parallel Server task on the Tasks tab, the following displays.



Complete the parameters on the tab and click the Submit button to run the Oracle Parallel Server startup task.

The Parameters tab for Shutdown contains these options:

Parameter	Description
Mode	Click the Immediate button (<i>default</i>) or the Abort button.
Connect As	Click the SYSDBA button (<i>default</i>) or the SYSOPER button.
Override Preferred Credentials	You can use the preferred credentials that have been set up for the database, or you can another database user name and password.

Monitoring Performance with Oracle Parallel Server Management Charts

This chapter presents the Oracle Performance Manager performance and tuning charts specific to Oracle8i Parallel Server. You must have Oracle Performance Manager installed and configured to display the charts. See [Chapter 5, "Installing and Configuring Oracle Parallel Server Management"](#) for further information.

This chapter is meant to only describe Oracle Performance Manager features specific to Oracle Parallel Server. Use this section as a supplement to general information contained in the *Oracle Enterprise Manager Performance Monitoring and Planning Guide*. This chapter covers the following topics:

- [Oracle Performance Manager Overview](#)
- [Starting the Oracle Performance Manager](#)
- [Displaying Charts](#)

See Also:

- ["Configuring Oracle Performance Manager"](#) on page 5-29, for configuration information.
- *Oracle8i Tuning* for information about the statistics these charts display, and how to interpret these statistics.
- *Oracle8i Reference* for more information about the fields in these charts and the V\$ views from which they are derived.

Oracle Performance Manager Overview

Tuning and performance information for the Oracle Parallel Server database is stored in a set of dynamic performance tables known as the "V\$ fixed views". Each active instance has its own set of fixed views. In Oracle Parallel Server, you can have Oracle Performance Manager essentially query a global dynamic performance (GV\$) view to retrieve the V\$ view information from all instances.

Oracle Performance Manager displays the retrieved information in a variety of tabular and graphic performance statistics for Oracle Parallel Server. The statistics represent the aggregate performance of all instances running on an Oracle Parallel Server. The statistics are displayed in individual charts and include information about data block pings, lock activity, file I/O, and session and user information. You can also use the Performance Manager to display an overview of all of these statistics on one chart.

Oracle Parallel Server performance monitoring is crucial for realizing the full potential of the system. There are several key performance metrics which need to be constantly monitored to keep the Oracle Parallel Server in peak operating condition. The Oracle Performance Manager, available as an applet within Oracle Enterprise Manager, is an application designed to capture, compute, and present performance data that help database administrators focus on key performance metrics.

Oracle Parallel Server performance metrics are compiled into charts that are viewable with Oracle Performance Manager:

Chart	Description
Parallel Server Block Ping Chart	Displays the total block pings on the Oracle Parallel Server.
Parallel Server Data Block Ping by Tablespace Chart	Displays the block pings per tablespace in the Oracle Parallel Server.
Parallel Server Data Block Ping by Instance Chart	Displays the block pings per instance in the Oracle Parallel Server.
Parallel Server File I/O Rate Chart	Displays the rate of physical reads and writes for all files in the Oracle Parallel Server database. You can drill down to obtain the same information either at the instance level or at the file level.
Parallel Server File I/O Rate by Object Chart	Displays the rate of reads and writes per data file in the Oracle Parallel Server database.

Chart	Description
Parallel Server File I/O Rate by Instance Chart	Displays the rate of reads and writes per instance in the Oracle Parallel Server database.
Parallel Server Lock Activity Chart	Displays the statistics on the lock activity rate for all the different lock types across all Oracle Parallel Servers. You can drill down to obtain lock activity information for a particular lock type at the instance level.
Parallel Server Sessions Chart	Displays the sessions attached to the Oracle Parallel Server and related information, such as instance name, session ID, session serial number, process ID, status, and user name.
Parallel Server Users Logged On Chart	Displays the total number of user sessions logged on to the Oracle Parallel Server, regardless of whether activity is generated. This information is also available for each instance.
Parallel Server Users Logged On by Instance Chart	Displays the number of users logged in to the Oracle Parallel Server by instance.
Parallel Server Active Users Chart	Displays the total number of active users on the Oracle Parallel Server.
Parallel Server Active Users by Instance Chart	Displays the number of active user sessions.
OPS Overview Chart	Displays a group of charts that display key performance statistics for the selected Oracle Parallel Server.

Starting the Oracle Performance Manager

To use the Oracle Enterprise Manager Console, start the following components:

- Oracle Data Gatherer on one of the nodes
- Oracle Performance Manager

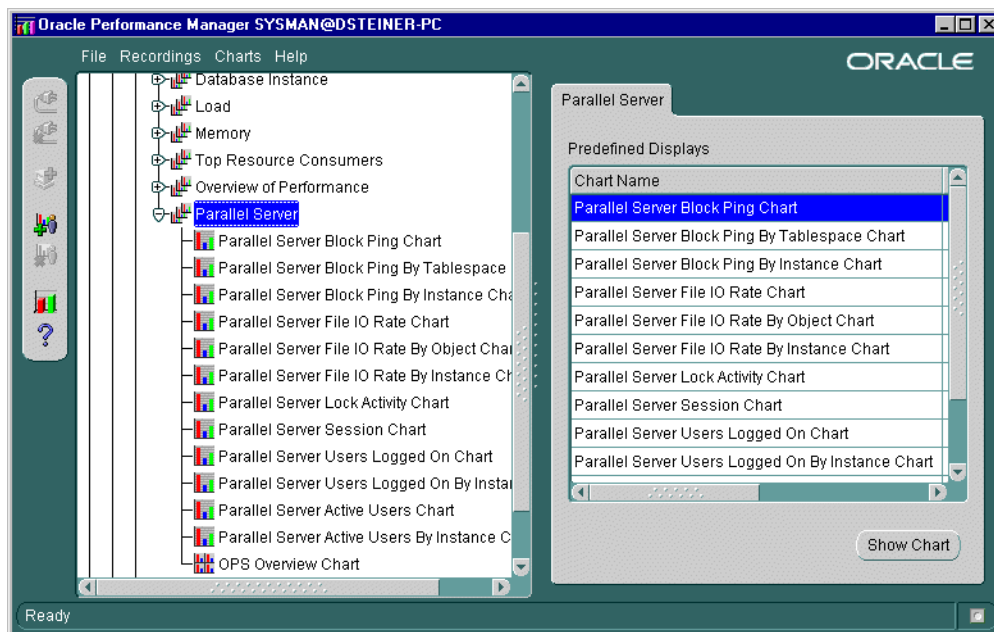
See "[Step 2: Start Oracle Performance Manager Components](#)" on page 5-30 for instructions.

Displaying Charts

To display charts:

1. Following the procedures in "[Step 2: Start Oracle Performance Manager Components](#)" on page 5-30 and "[Step 3: Accessing Oracle Parallel Server Charts](#)" on page 5-34.

- From within Oracle Performance Manager, double-click Parallel Server from the Databases or Parallel Server Instances folder to see the list of available charts.



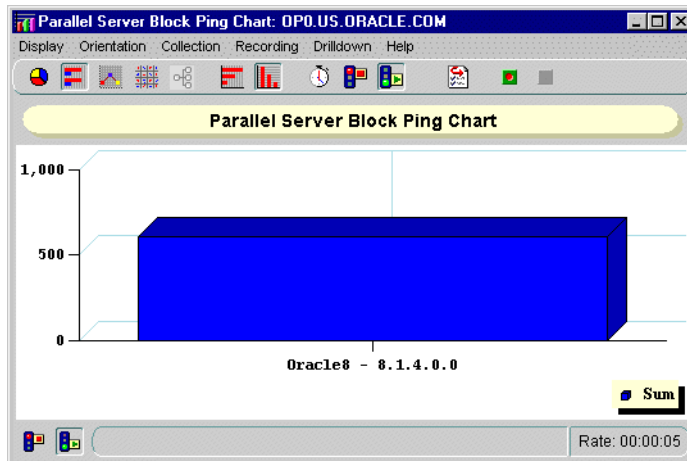
- From within the Parallel Server object folder, select a chart, then click Show Chart.

The chart displays in separate window.

Parallel Server Block Ping Chart

The Parallel Server Block Ping chart displays statistics from the GV\$PING view on the total number of block pings:

Figure 7-1 Parallel Server Block Ping Chart



To display block pings at the tablespace or instance level:

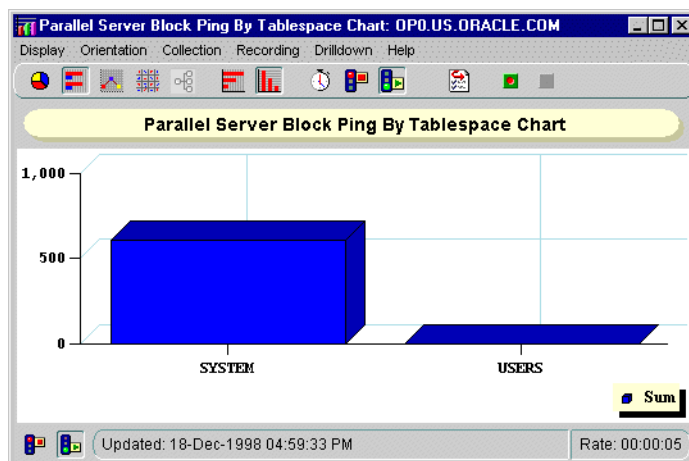
1. Click anywhere on the chart.
2. Right-click and select the chart from the Drilldown menu.

Chart	Description
Parallel Server Data Block Ping by Instance Chart	Displays pinging on individual instances.
Parallel Server Data Block Ping by Tablespace Chart	Displays the pinging on the tablespaces.

Parallel Server Data Block Ping by Tablespace Chart

The Parallel Server Block Ping by Tablespace chart displays statistics from the GV\$PING view on the number of block pings by tablespace:

Figure 7-2 Parallel Server Block Ping by Tablespace Chart



To display block pings at the object or instance level:

1. Click on an individual tablespace.
2. Right-click and select the chart from the Drilldown menu:

Option	Description
Parallel Server Block Ping Tablespace Object Chart	Displays the objects for the selected tablespace. Objects represent tables, indexes, free extents, clusters of tables, and so on. Listed for each object is the number of pings in the tablespace in which that object resides.
Parallel Server Block Ping Tablespace Instance Chart	Displays the instances within the selected tablespace. Listed for each instance is the number of pings in the tablespace in which that instance resides.

Figure 7-3 OPS Block Ping Tablespace Object Chart

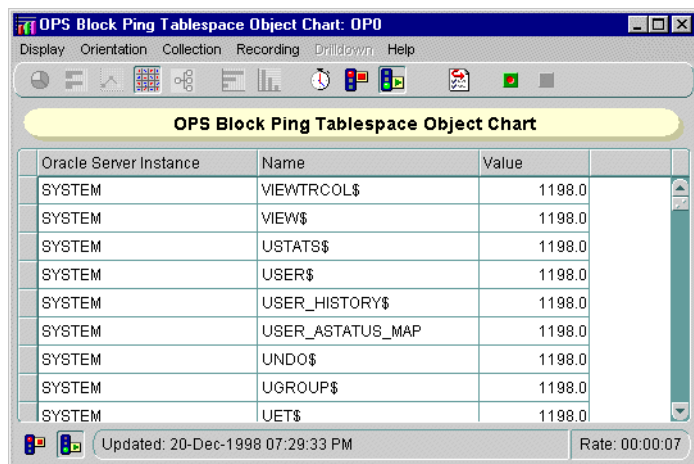
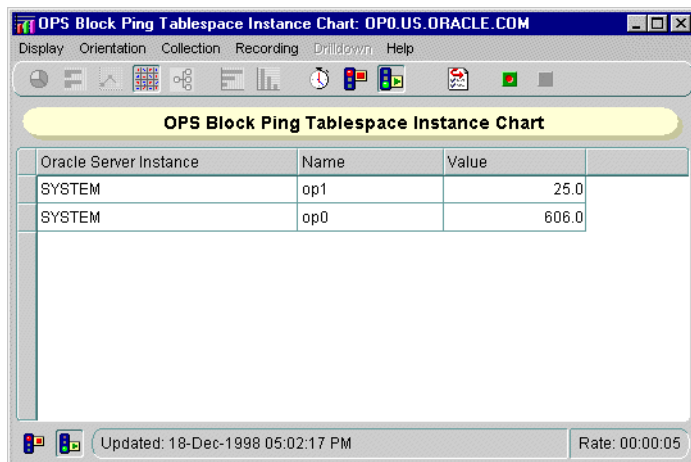


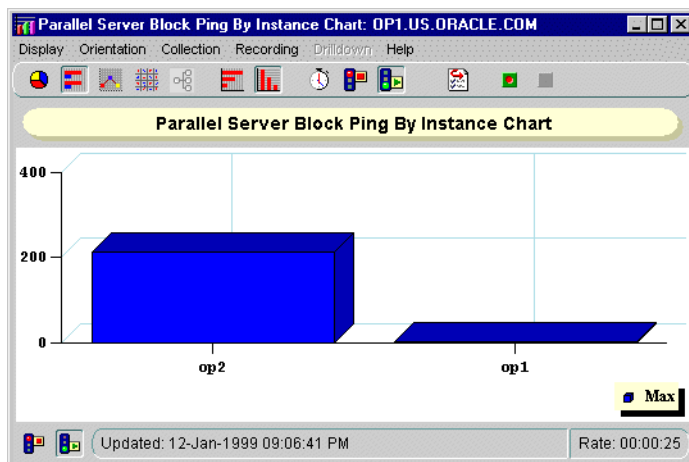
Figure 7-4 OPS Block Ping Tablespace Instance Chart



Parallel Server Data Block Ping by Instance Chart

The Parallel Server Block Ping by Instance chart displays statistics from the GV\$PING view on the number of block pings per instance:

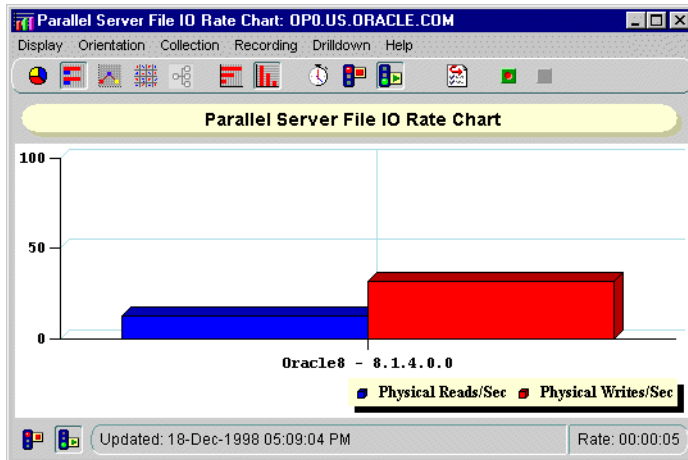
Figure 7-5 *Parallel Server Block Ping by Instance Chart*



Parallel Server File I/O Rate Chart

The Parallel Server File I/O Rate chart displays physical files reads and writes for all files from the GV\$FILESTAT view.

Figure 7-6 Parallel Server File IO Rate Chart



To display the file I/O rate at the instance level or at the file level:

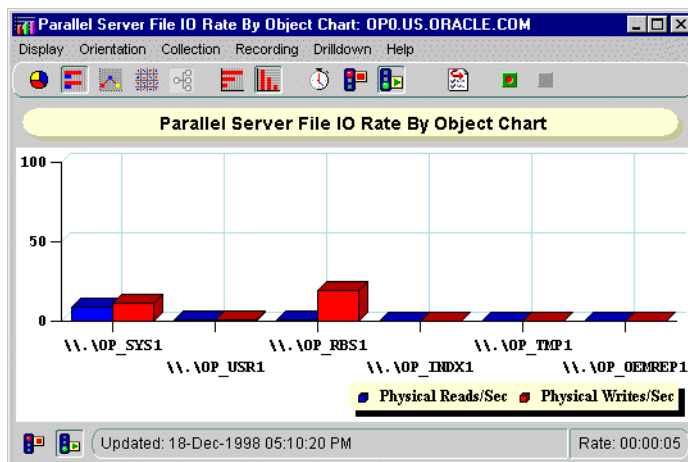
1. Click on a column's object.
2. Right-click and select the chart from the Drilldown menu.

Chart	Description
Parallel Server File I/O Rate by Instance Chart	Displays the file I/O rate on individual instances.
Parallel Server File I/O Rate by Object Chart	Displays the file I/O rate on the tablespaces.

Parallel Server File I/O Rate by Object Chart

The Parallel Server File I/O Rate by Object chart displays physical files reads and writes for individual data files from the GV\$FILESTAT view.

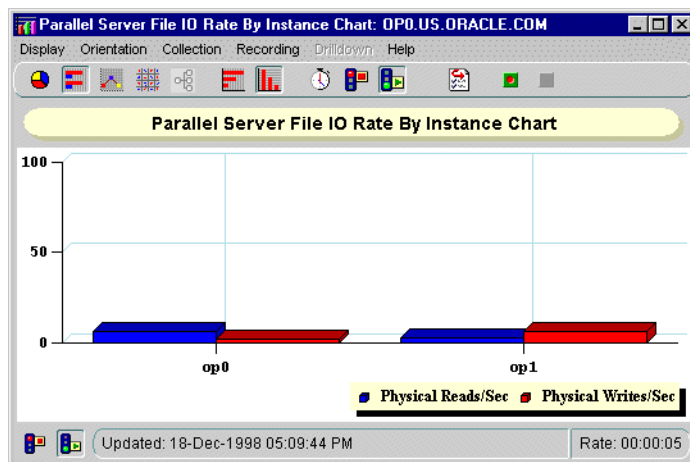
Figure 7-7 Parallel Server Block Ping by Object Chart



Parallel Server File I/O Rate by Instance Chart

The Parallel Server File I/O Rate by Instance chart displays physical files reads and writes for individual instances from the GV\$FILESTAT view.

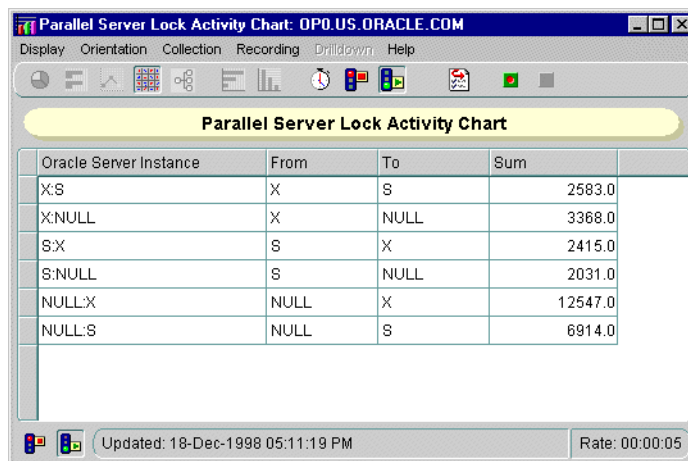
Figure 7–8 *Parallel Server File IO Rate by Instance Chart*



Parallel Server Lock Activity Chart

The Parallel Server Lock Activity chart shows the number of lock converts of various types (exclusive to null) from the GVSLOCK_ACTIVITY view:

Figure 7–9 Parallel Server Lock Activity Chart

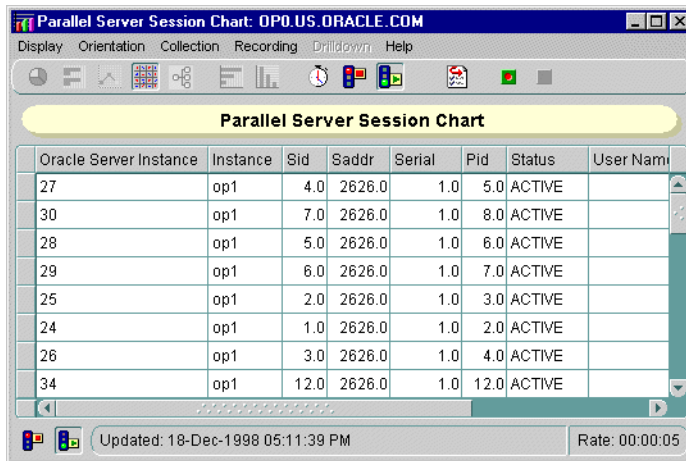


Column	Description
From	PCM lock initial state: NULL; S; X; SSX
To	PCM lock final state: NULL; S; X; SSX
Sum	Number of times the lock operation executed

Parallel Server Sessions Chart

The Parallel Server Sessions chart displays a table of Session IDs. This chart displays statistics from the GV\$PROCESS and GV\$SESSION and GV\$PROCESS views.

Figure 7-10 Parallel Server Sessions Chart

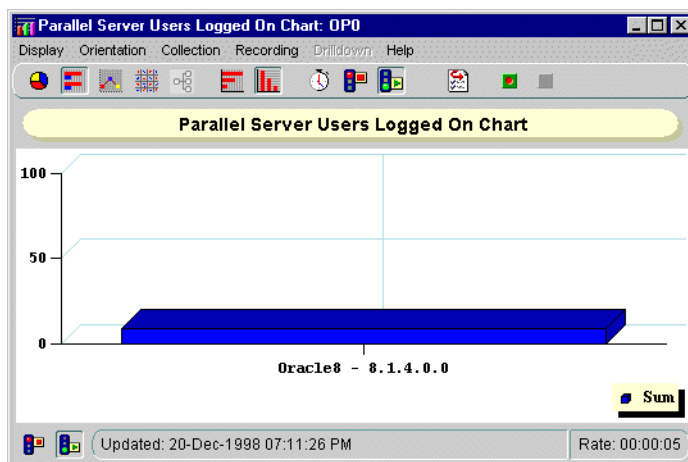


Column	Description
Sid	Session identifier
Saddr	Session address
Serial	Session serial number. Used to identify uniquely a session's objects. Guarantees that session-level commands are applied to the correct session objects if the session ends and another session begins with the same session ID.
PID	Operating system client process ID
Status	Status of the session: ACTIVE (currently executing SQL), INACTIVE, KILLED (marked to be killed), CACHED (temporarily cached for use by Oracle*XA), SNIPED (session inactive, waiting on the client)
User Name	Oracle user name

Parallel Server Users Logged On Chart

The Parallel Server Users Logged On chart displays the total number of user sessions currently logged on to the Oracle Parallel Server, whether or not activity is being generated. This chart displays statistics from the GV\$LICENSE view:

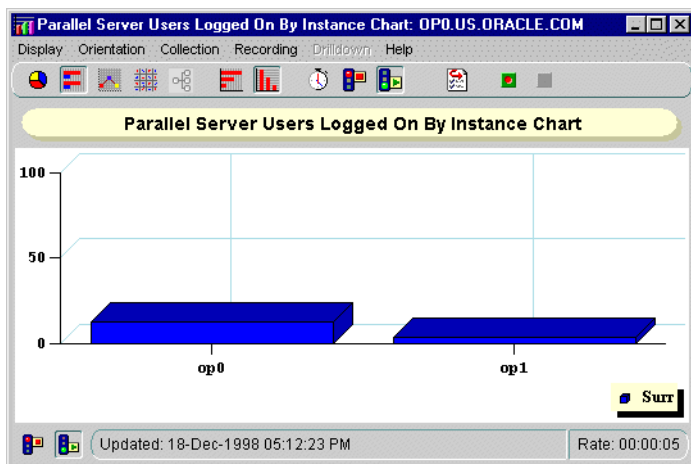
Figure 7-11 Parallel Server Users Logged On Chart



Parallel Server Users Logged On by Instance Chart

The Parallel Server Users Logged On by Instance chart displays the number of users logged on to each instance in the Oracle Parallel Server. This chart displays statistics from the GV\$LICENSE view.

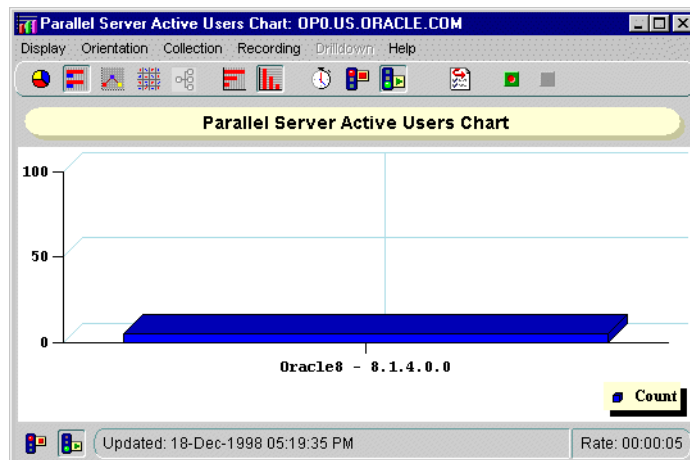
Figure 7-12 *Parallel Server Users Logged On by Instance Chart*



Parallel Server Active Users Chart

The Parallel Server Active Users chart displays the total number of user active sessions. This chart displays statistics from the GVSLICENSE view:

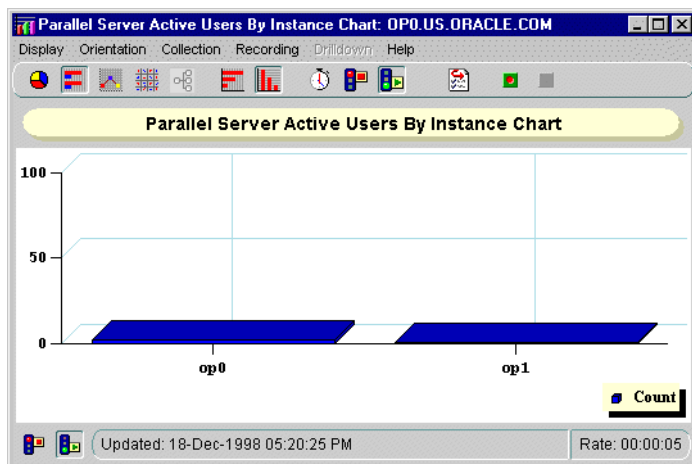
Figure 7–13 Parallel Server Active Users Chart



Parallel Server Active Users by Instance Chart

The Parallel Server Active Users by Instance chart displays the number of active user session logged on to each instance in the Oracle Parallel Server. This chart displays statistics from the GV\$LICENSE view.

Figure 7-14 *Parallel Server Active Users by Instance Chart*

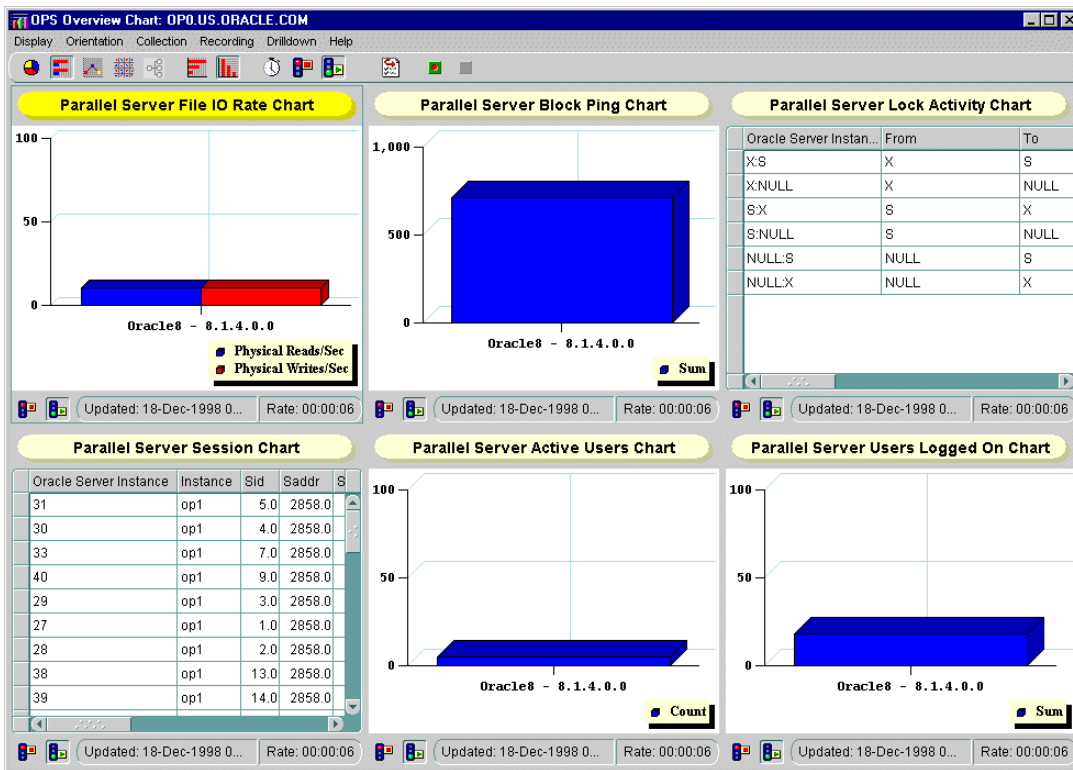


OPS Overview Chart

The OPS Overview chart displays the following charts:

- Parallel Server File I/O Rate Chart
- Parallel Server Block Ping Chart
- Parallel Server Lock Activity Chart
- Parallel Server Users Logged On Chart

Figure 7-15 Overview Chart



Adding Instances and Nodes

This chapter describes how to convert from a single instance Oracle8i database to a multi-instance Oracle8i database using the Oracle Parallel Server Option, and how to add a additional node to an existing xx-node configuration.

Specific topics discussed are:

- [Moving from a Single Instance to Parallel Server](#)
- [Adding Additional Nodes to a Cluster](#)

Moving from a Single Instance to Parallel Server

This section explains how to migrate from a single instance Oracle[®]8i database to multi-instance. In addition, it can help you extend an existing Oracle Parallel Server configuration to additional nodes.

The steps to migrate from a single instance to an Oracle Parallel Server include:

[Step 1: Configure Hardware](#)

[Step 2: Evaluate Tablespaces and Log Files of Single Instance](#)

[Step 3: Create Raw Devices](#)

[Step 4: Export Data from Old Database](#)

[Step 5: Install Operating System Dependent Files](#)

[Step 6: Install Oracle[®]8i Enterprise Edition and Oracle[®]8i Parallel Server Option](#)

[Step 7: Create the Database](#)

[Step 8: Transfer Data From Old Data to Empty Database](#)

[Step 9: Start the Database](#)

This section should be used as a supplement to Chapter 23, "Migrating from Single Instance to Parallel Server," of the *Oracle[®]8i Parallel Server Concepts and Administration* guide.

Step 1: Configure Hardware

See your vendor documentation for information about setting up Oracle Parallel Server hardware.

Step 2: Evaluate Tablespaces and Log Files of Single Instance

Because tablespaces must be identical on the Oracle Parallel Server as a single instance database, it is important that you do any consolidation, addition of, or renaming of tablespaces on the single instance at this time.

Each additional node in the cluster requires two log files. Typically, a single instance database only has two log files. You must add additional log files for each node using the ALTER DATABASE ADD LOGFILE statement.

Step 3: Create Raw Devices

Create the raw devices that will be needed for the data files, control files and log files needed for Oracle Parallel Server, as explained in "[Setting Up Raw Devices](#)" on page 2-6.

Step 4: Export Data from Old Database

Export the entire database from the single instance database. Use a tool such as the Export utility

To use the Export utility to export all data from an existing database to the new database:

Note: To export an entire database, you must use the user name SYSTEM. Do not use INTERNAL or SYS.

Run the Export utility:

```
exp username/password file=file.dmp full=y log file.log
```

FILE.DMP represents the data from the full database export of the database, while *FILE.LOG* represents the log of the operation.

See *Oracle8i Utilities* for further information about this tool.

Step 5: Install Operating System Dependent Files

For all nodes, see your Operating System Dependent vendor documentation for instructions about installing Operating System Dependent files.

Step 6: Install Oracle8i Enterprise Edition and Oracle8i Parallel Server Option

Note: The Custom install type is only supported on Windows NT.

Except on the server already running Oracle8i Enterprise Edition, install Oracle8i Enterprise Edition along with the Oracle Parallel Server option, as described in [Chapter 3, "Installation"](#). If your current single instance database does not have the supporting hardware, you must perform a clean install on new cluster hardware.

On the server already running Oracle8i Enterprise Edition, install the Oracle Parallel Server option using the Custom installation type.

Step 7: Create the Database

A new database must be created on the raw partitions. Depending on the install type and subsequent configuration options, a database may be created with the following methods:

- If you selected the Typical install type, the database will be created after installation.
- If you requested the Oracle Database Configuration Assistant run after the Custom install type, the Oracle Database Configuration creates a database.
- If you requested the Oracle Database Configuration Assistant not run after during Custom install type, you can still run the Oracle Database Configuration to create a database. See ["Creating the Database"](#) on page 4-3.
- If you requested the Oracle Database Configuration Assistant not run after during Custom install type, you can create the database manually.

Step 8: Transfer Data From Old Data to Empty Database

Import the entire database into the empty database. Use a tool such as the Import utility.

To run the Import utility:

```
imp system/password file=file.dmp full=y log file.log
```

FILE.DMP represents the data from the full database export of the database, while *FILE.LOG* represents the log of the operation.

See *Oracle8i Utilities* for further information about this tool.

ATTENTION: If the original database from which the export file was generated contains tablespaces that are not in the new database, then the Import utility attempts to create those tablespaces with associated data files. The easy solution is to ensure both databases contain the same tablespaces. The data files do not have to be identical. Only the tablespace names need to be the same.

Step 9: Start the Database

See "[Starting the Database in Parallel Mode](#)" on page 4-9 to start the Oracle Parallel Server.

Adding Additional Nodes to a Cluster

If you have already configured a *xx*-node cluster and want to add other nodes, Oracle Corporation recommends the follow procedure:

1. Install another database into a different Oracle home that includes all the nodes you want the to support, as described in [Chapter 3, "Installation"](#).
2. Use the Oracle Database Configuration Assistant to configure the database for all the nodes, as described in [Chapter 3, "Installation"](#) and ["Creating the Database"](#) on page 4-3.
3. Export the data from the existing database, as described in ["Step 4: Export Data from Old Database"](#) on page 8-3.
4. Import the data to the new database, s described in ["Step 8: Transfer Data From Old Data to Empty Database"](#) on page 8-5.

A

Directory Structure

Specific topics covered in this appendix are:

- [Understanding the Oracle Parallel Server Directory Structure](#)
- [UNIX](#)
- [Windows NT](#)

Understanding the Oracle Parallel Server Directory Structure

When you install Oracle8i Enterprise Edition and the Oracle Parallel Server Option, all subdirectories are under a top-level ORACLE_BASE. ORACLE_HOME and admin directories are also located under ORACLE_BASE.

UNIX

The following is the hierarchical directory tree of a sample OFA-compliant database for Oracle Parallel Server on UNIX platforms:

\$ORACLE_		/u01/app/oracle is the default ORACLE_BASE
BASE		directory
	\$ORACLE_HOME	/product/8.1.5 is the name of the Oracle home by default
	/bin	Subtree for Oracle binaries
	/network	Subtree for Net8
	/ops	Files necessary for Oracle Database Configuration Assistant
	/opsm	Oracle Parallel Server Management message and TCL files
	/dbs	This is a legacy directory from previous releases. It contains links to initialization files that point to the new location of the initialization parameter files, \$ORACLE_BASE/admin/db_name/pfile.
	...	
	/admin	Subtree for Oracle Parallel Server database administration files
	/db_name	Database administration files for this database identified by the database name
	/adhoc	Ad hoc SQL scripts
	/adump	Audit files
	/arch	Archived redo log files
	/bdump	Background process trace files
	/cdump	Core dump files
	/create	Programs used to create the database
	/exp	Database export files
	/pfile	Initialization parameter files
	/udump	User SQL trace files

For more information about the contents of the \$ORACLE_HOME and /admin directories, see *Oracle8i Administrator's Reference* for Sun Solaris, HP 9000 or AIX-based systems.

Windows NT

The following is the hierarchical directory tree of a sample OFA-compliant database for Oracle Parallel Server on Windows NT:

<i>x:\oracle_</i>		<i>c:\oracle</i> is the default ORACLE_BASE directory
<i>base</i>		
	<i>\oracle_home</i>	<i>\ora81</i> is the name of the Oracle home by default
		Subtree for Oracle binaries
	<i>\bin</i>	
	<i>\network</i>	Subtree for Net8 configuration files, including TNSNAMES.ORA, LISTENER.ORA and SQLNET.ORA
	<i>\ops</i>	*.TBL sample files to create raw partitions for the Oracle Database Configuration Assistant
	<i>\database</i>	This is a legacy directory from previous releases. It contains initialization files that points to the new directory location for the initialization parameter files, <i>oracle_base\db_name\pfile</i> .
	...	
	<i>\admin</i>	Subtree for Oracle Parallel Server database administration files
	<i>\db_name</i>	<i>DB_NAME</i> database administration files for the instance identified by <i>SID</i> .
	<i>\ad hoc</i>	Ad hoc SQL scripts
	<i>\adump</i>	Audit files
	<i>\arch</i>	Archived redo log files
	<i>\bdump</i>	Background process trace files
	<i>\cdump</i>	Core dump files
	<i>\create</i>	Programs used to create the database
	<i>\exp</i>	Database export files
	<i>\pfile</i>	Initialization parameter files
	<i>\udump</i>	User SQL trace files

For more information about the contents of the ORACLE_HOME and \admin directories, see *Oracle Parallel Server Getting Started for Windows NT*.

Oracle Parallel Server Management on UNIX Reference

This appendix describes Oracle Parallel Server Management on UNIX.

Specific topics covered in this appendix are:

- [OPCTL Utility](#)
- [Creating the DB_NAME.CONF File](#)
- [Oracle Intelligent Agent Configuration](#)

OPSCTL Utility

The OPSCTL utility on UNIX uses a manually-created file named *DB_NAME.CONF* to define Oracle Parallel Server instances and related services.

Important: The *DB_NAME.CONF* file must reside in the Oracle Parallel Server's shared or non-shared Oracle home location, even if this file is empty. OPSCTL requires this file to run properly.

Consider the following points when configuring your *DB_NAME.CONF* file:

- The `$ORACLE_HOME` location can be set up to be shared or non-shared. If the location is non-shared, you must manually create and FTP the *DB_NAME.CONF* file to each node of the cluster.
- The managed system maps the names of the nodes to correspond with non-negative numbers (positive numbers) called "node numbers" based on the thread ID.

Creating the *DB_NAME.CONF* File

This section covers the following topics:

- [Parameter Syntax](#)
- [Parameter Expansion](#)
- [Parameter Descriptions](#)
- [Parameter Node Overrides](#)
- [Oracle Parallel Server Instances](#)

Parameter Syntax

The parameter values are specified in the configuration file using the following syntax:

```
parameter_name = value
parameter_name = (value1, value2, ...)
```

Consider these points when configuring parameter values:

- Each parameter must be entered on a separate line in the *DBNAME.CONF* file.
- Lists of values are comma-separated and enclosed within parenthesis.
- Any parameter value that contains commas or spaces must be surrounded by double quotation marks (“xxx”).

Parameter Expansion

Parameter values with the percent sign (%) in the configuration file have different meanings depending on the character that follows the % sign:

Value	Description
%p	Expands to the name of the Oracle Parallel Server which is set by the value of ORACLE_PSRV parameter or the environment variable ORACLE_PSRV.
%h	Expands to the value of the ORACLE_HOME parameter, if set, or expands to the value of the ORACLE_HOME.
%n	Expands to the node number for the appropriate node. The %n is replaced by a different value on each node.
%m	Expands to the node name for the appropriate node. The %m is replaced by a different value on each node.

Parameter expansion examples are shown in the table below:

Example	Description
inst_init_ora=/ora/dbs/initop.ora	The name of the instance parameter file is INITOP.ORA. This parameter file applies to all nodes of the Oracle Parallel Server; there is no parameter expansion.
inst_init_ora=/ora/dbs/initop_ \ %m.ora	This example, containing initop_%m.ora, illustrates parameter expansion which is replaced by a different value on each node. For example, if there are three nodes with the names spdev01, spdev02, and spdev03, the parameter expansion values for these nodes with node_list = "1-3" are as follows: initop_spdev01.ora initop_spdev02.ora initop_spdev03.ora

Parameter Descriptions

The following table lists the configuration parameters that may be set in the *DB_NAME.CONF* file. Each parameter is listed in the order that it would typically appear in the configuration file. Parameters must be specified in lowercase.

Parameter	Description
<code>node_list</code>	<p>The list of node numbers for all nodes on which Oracle Parallel Server instances are run. This parameter is required on some platforms, including IBM RS/6000 SP2.</p> <p>On other platforms, the entire cluster is used by default.</p> <p>Example: <code>node_list = "1-8"</code></p>
<code>inst_oracle_sid</code>	<p><i>(optional)</i> The instance identifier. Each Oracle Parallel Server requires a unique Oracle System Identifier (SID). The default is <code>%p%n</code>.</p>
<code>inst_init_ora</code>	<p><i>(optional)</i> A parameter file that is shared by all the instances of the Oracle Parallel Server. If the parameter contains <code>%n</code> or <code>%m</code>, a different <code>INITSID.ORA</code> on each node is used.</p>
<code>node#:inst_init_ora</code>	<p><i>(optional)</i> A parameter file for the Oracle Parallel Server instance on a specific node, as denoted by the <i>node#</i>. This overrides any value specified in the <code>inst_init_ora</code> parameter.</p>
<code>oracle_home</code>	<p><i>(optional)</i> Specifies either the single, shared Oracle home or each non-shared Oracle home location for each node. Defaults to the Oracle home specified in <code>oratab</code> (platform-specific location).</p>
<code>lsnr_listener_name</code>	<p><i>(optional)</i> The names of the listeners for each specific node.</p>
<code>tns_admin</code>	<p><i>(optional)</i> The value of the <code>TNS_ADMIN</code> environment variable which is used to start or stop all listeners.</p>
<code>node#:tns_admin</code>	<p><i>(optional)</i> The value of <code>TNS_ADMIN</code> environment variable which is used to start or stop the listener on the specific node, as denoted by <i>node#</i>.</p>

Parameter	Description
environ	<p>(<i>optional</i>) The double-quoted list of semi-colon separated "<i>name = value</i>" pairs of environment variables that are required for the startup or shutdown operations.</p> <p>For example:</p> <pre>environ="ENV1=value1;ENV2=value2"</pre> <p>The environment variables included in this parameter are in addition to ORACLE_PSRV, ORACLE_HOME, and TNS_ADMIN, each of which are represented either by a separate environment variable or a separate configuration parameter.</p>

Shown below are examples of how the configuration parameters are applied. The parameter settings for different services are illustrated with reference to a fictitious Oracle Parallel Server, named MYPSRV, on a fictitious eight-node system. The following table shows how node numbers, node names, and instance identifiers are mapped to each other for MYPSRV. Refer to this table as required.

Node Number	Node Name	SID
1	spdev01	SID_A
2	spdev02	SID_B
3	spdev03	SID_C
6	spdev06	SID_D
7	spdev07	SID_E
8	spdev08	SID_F

Parameter Node Overrides

You can set the parameters to apply to a specified set of nodes by prefixing the parameter definition with a node number as follows:

```
node_number:parameter_name =value
```

Oracle Parallel Server Instances

Instances are defined by the following parameters:

- [inst_oracle_sid](#)
- [inst_init_ora](#)

inst_oracle_sid

If the `inst_oracle_sid` parameter is not specified, then OPCTL generates a default list of Oracle System Identifiers (SID) in which each SID is set to the following format:

```
dbname node#
```

where *dbname* corresponds to the name of the Oracle Parallel Server and where *node#* corresponds to the node number of the corresponding node in the `NODE_LIST`. This is as if `%p%n` were specified.

For example, if the parameters are set as follows:

```
oracle_psrv = MYPSRV
node_list = "1-3,6"
```

and the `inst_oracle_sid` parameter is not specified, then the SIDs generated for the instances will be MYPSRV1, MYPSRV2, MYPSRV3, and MYPSRV6 for the `spdev01`, `spdev02`, `spdev03`, and `spdev06` nodes respectively.

The `inst_oracle_sid` parameter is used to specify the `ORACLE_SIDs` (instance identifiers) for the instances. It may be a comma-separated list of SID names where each SID maps to the corresponding node mentioned in the same order in the `node_list` parameter. Or, it may be that `%n` or `%m` is used to generate a distinct SID on each node.

For example, if the parameters are set as follows:

```
node_list = "1-3,6"
inst_oracle_sid = (SID_A, SID_B, SID_C, SID_D)
```

then, the `SID_A` value will be used for the instance on node `spdev01` and the `SID_D` value will be used for the instance on the `spdev06` node.

The parameter substitution facility can be used to customize `ORACLE_SID` values. For example, if the parameters are set as follows:

```
node_list = "1-3,6"  
inst_oracle_sid = "%p_SID_%n"
```

then, the `MYPSRV_SID_1`, `MYPSRV_SID_2`, `MYPSRV_SID_3`, and `MYPSRV_SID_6` values are used for SIDs of instances on nodes `spdev01`, `spdev02`, `spdev03`, and `spdev06` respectively.

inst_init_ora

The absolute path names for the parameter files are specified by the `inst_init_ora` and/or `node#:inst_init_ora` parameters. If this parameter is not specified, then `$ORACLE_HOME/dbs/initsid.ora` is used.

For each instance, OPCTL first searches for the `node#:inst_init_ora` parameter, where `node#` is the number of the node on which the instance runs. If this parameter is set, OPCTL uses its value as the instance's parameter file. If it is not specified, then OPCTL uses the `inst_init_ora` parameter.

Listeners

Listeners are defined by the following parameters:

- `tns_admin`
- `lsnr_listener_name`

`tns_admin`

The value for `tns_admin` for all listeners to be started or stopped can be set with the `tns_admin` parameter.

If the `tns_admin` parameter is not set and the `TNS_ADMIN` environment variable is set, the listeners are given the value of the `TNS_ADMIN` environment variable.

If neither the `tns_admin` parameter nor the `TNS_ADMIN` environment variable are set, the listeners are not given any value, and will use the default search order to locate the `LISTENER.ORA` file.

The `LISTENER.ORA` file contains the listening addresses of the listener on the system as well as the name and Oracle home of any database the listener serves.

To use a different `TNS_ADMIN` variable for a specific node, you may specify the node using the `node#:tns_admin` parameter, where `node#` refers to the number of the corresponding node.

For example, if the `TNS_ADMIN` environment variable is set to `/home/myname/misc` and the `tns_admin` and none of the `node#:tns_admin` parameters are specified in the configuration file, then `OPCTL` uses the value `/home/myname/misc` for all nodes.

In another example, if the following parameters are set as follows:

```
node_list = "1-3,6"
tns_admin = /dve/myname/815/ops
3:tns_admin = /dve/myname/others
```

then, for the listener on `spdev03` (node number 3), `tns_admin` is set to `/dve/myname/others`, whereas nodes `spdev01`, `spdev02` and `spdev06`, `tns_admin` is set to `/dve/myname/815/ops`.

lsnr_listener_name

The `lsnr_listener_name` parameter is used to specify the listener names. It is a comma-separated list of listener names, where each listener name maps to the corresponding node mentioned in the same order listed for the `node_list` parameter.

If the `lsnr_listener_name` parameter is not specified, then OPSCCTL generates a default list of listener names in which each listener name follows the format `node_name_listener`, where `node_name` corresponds to the name of the node. This is as if `%m_listener` was specified.

For example, if the `node_list` parameter is set as follows:

```
node_list = "1-3,6"
```

and the `lsnr_listener_name` parameter is not specified, then the names generated for the listeners would be `spdev01_listener`, `spdev02_listener`, `spdev03_listener`, and `spdev06_listener` for the `spdev01`, `spdev02`, `spdev03`, and `spdev06` nodes respectively.

For example, if the parameters are set as follows:

```
node_list = "1-3,6"
lsnr_listener_name = (listener_A, listener_B, listener_C, listener_D)
lsnr_listener_name = (listener_A, listener_B, listener_C, listener_D)
```

then, the `listener_A` name would be used for the listener on node `spdev01` and the `listener_D` name would be used for the listener on node `spdev06`.

If only one entry is specified for the `lsnr_listener_name` parameter, then the same name is used for listeners on all nodes specified in the `node_list`. Also, you can use parameter expansion to specify different listener names for all nodes.

For example, if the parameters are set as follows:

```
node_list = "1-3,6"
lsnr_listener_name = "listener_myprsv"
```

then, the `listener_myprsv` name would be used for all the listeners on all nodes.

In another example, if the following parameters are set as follows:

```
node_list = "1-3,6"
lsnr_listener_name = "lsnr_%n"
```

then, the names `lsnr_1`, `lsnr_2`, `lsnr_3`, and `lsnr_6` will be used for the listeners on the nodes `spdev01`, `spdev02`, `spdev03`, and `spdev06` respectively.

Sample of *DB_NAME.CONF* File

The following is an example *DB_NAME.CONF* file:

```
node_list = "1-4"
inst_init_ora = /ora/dbs/init_%m.ora
inst_oracle_sid = (SID_A, SID_B, SID_C, SID_D)
lsnr_listener_name = listener_myhost%n
```

Oracle Intelligent Agent Configuration

Oracle Intelligent Agents are shipped with the database and must be installed on each remote, managed Oracle Parallel Server node. The Oracle Intelligent Agents are responsible for discovering the available services (databases, listeners, parallel servers) on each node.

The configuration is different for shared and non-shared ORACLE_HOME locations. In a typical setup, the Oracle Intelligent Agents are installed in non-shared ORACLE_HOME locations.

The following directories should *not* be shared because they contain files that are specific to each Oracle Intelligent Agent's node:

- \$ORACLE_HOME/network/admin
- \$ORACLE_HOME/network/log
- \$ORACLE_HOME/network/trace
- \$ORACLE_HOME/network/agent

CAUTION: The Oracle Intelligent Agent cannot be installed in a shared Oracle home location. You must ensure that each node's Oracle Intelligent Agent is installed in a separate Oracle home location, distinct from other Oracle home locations.

Also, ensure that your server files, TNSNAMES.ORA and LISTENER.ORA, contain the correct entries for all instances to perform auto-discovery. See "[Configuring Net8 for Nodes](#)" on page 4-12 for further information.

Non-shared Oracle Home

You must ensure that you configure and start the Oracle Intelligent Agents on every node running an instance.

To start the agent, at an operating system prompt, enter the command:

```
lsnrctl dbsnmp_start
```

Shared Oracle Home

If the Oracle home location is shared (through NFS or any other equivalent facility) by all nodes, you must set up the Oracle Intelligent Agent as follows:

1. Install the Oracle Intelligent Agent for each node in its own Oracle home location, distinct from the shared Oracle home location.

Note: You cannot install the Intelligent Agent in the same shared Oracle home location where your Oracle database is located.

2. Copy or link TNSNAMES.ORA and LISTENER.ORA to each Oracle Intelligent Agent's Oracle home location from the shared Oracle home.
3. Before starting the Oracle Intelligent Agent on each node, set the Oracle home location to the Oracle Intelligent Agent's Oracle home.
4. Issue the `lsnrctl dbsnmp_start` command.
5. Ensure that the DBSNMP utility is set up to run at system startup time.

C

Troubleshooting

Specific topics covered in this appendix are:

- [Cleaning Up the Registry After an Oracle Database Configuration Assistant Failure on Windows NT](#)
- [Handling Auto-Discovery Failures](#)
- [Using Trace Files](#)
- [Contacting Oracle Worldwide Customer Support](#)

Cleaning Up the Registry After an Oracle Database Configuration Assistant Failure on Windows NT

If the Oracle Database Configuration Assistant fails during the creation of a database on Windows NT, certain entries may have been installed in the registry. When a database fails:

1. Delete the *DB_NAME* sub-key in the Performance and Management (P&M) key in the registry on *all* nodes prior to running the application again:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\PM
\DB_NAME
```

If you do not delete this key, the Oracle Database Configuration Assistant will assume the *db_name* database name is already in use. The next time you run the Oracle Database Configuration Assistant, you will not be able to use this database name, even though the raw devices will expect it.

2. Delete the key for OracleService*SID* key in the registry on the node from which the Oracle Database Configuration Assistant was run:

```
HKEY_LOCAL_
MACHINE\SYSTEM\CURRENTCONTROLSET\SERVICES\OracleServiceSID
```

Handling Auto-Discovery Failures

Auto-discovery of nodes or objects in the Console typically fails because Oracle Intelligent Agent was not being started on the node or the configuration is incorrect. If starting the Oracle Intelligent Agent does not resolve the problem, it may be a more serious configuration issue.

This section covers the following topics:

- [Understanding Auto-Discovery](#)
- [Auto-Discovery Results](#)
- [Troubleshooting Auto-Discovery](#)

Understanding Auto-Discovery

To understand proper configuration, it is important to understand how auto-discovery works. During the auto-discovery process, a SERVICES.ORA file on the managed nodes is created in the `$ORACLE_HOME/network/agent` directory on UNIX platforms and `ORACLE_HOME\network\admin` directory on Windows NT. This file contains information about the nodes and their services (databases, instances and listeners) discovered.

This file is created from the following sources, all on the managed nodes:

- [oratab on UNIX and Registry on Windows NT](#)
- [LISTENER.ORA](#)
- [TNSNAMES.ORA](#)

Each of these components must be configured correctly in order for auto-discovery to work correctly.

oratab on UNIX and Registry on Windows NT

Auto-discovery first discovers the Oracle Parallel Server database name and the nodes associated with the database. How it accomplishes this is dependent on whether the managed system is running on:

- UNIX
- Windows NT

UNIX On UNIX platforms, auto-discovery uses information in the `oratab` entry for the name of the Oracle Parallel Server. `oratab` is found in `/etc/oratab` or `/var/opt/oracle/oratab`. It contains entries of the form:

```
db_name: oracle_home:N
```

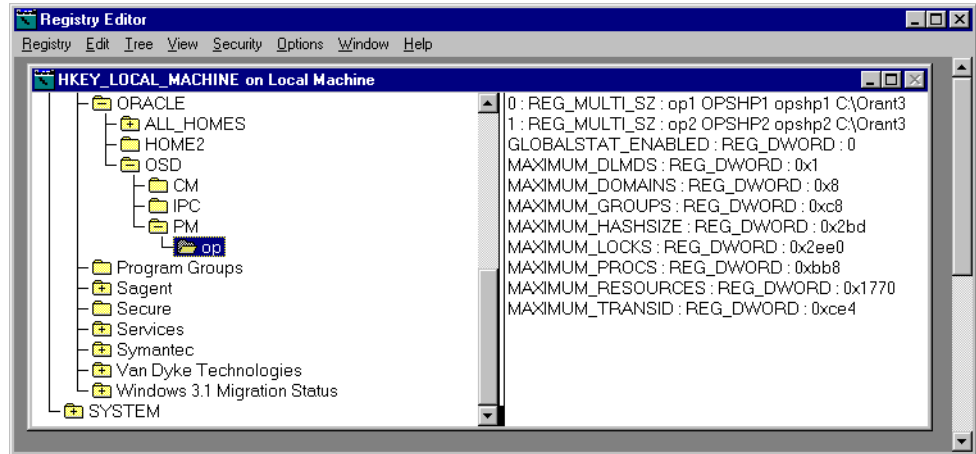
where `db_name` is the database name and `oracle_home` is the Oracle home given to your Oracle Parallel Server database. From this entry, the database name is acquired.

Next, auto-discovery looks for the `node_list` parameter in `DB_NAME.CONF` file, located in `$ORACLE_HOME/ops`, to determine which nodes run instances of the Oracle Parallel Server:

```
node_list = "1,2,3"
```

On some platforms `node_list` defaults to the entire cluster, and this parameter does not need to be explicitly set. The configuration file must exist even if it is empty.

Windows NT The registry lists all Oracle Parallel Servers that run on a node under HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\PM. Under this subkey, each Oracle 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.

The following are the registry entries required for auto-discovery:

Value	Value Type	Description
		HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\PM
		HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD\PM\DB_NAME
0	REG_MULTI_SZ	Specifies the cluster instance ID data assigned to the OP1 on the first node with the following format: <i>SID COMPUTER_NAME HOST_NAME ORACLE_HOME</i> op1 OPSHP1 opshp1 c:\orant
1	REG_MULTI_SZ	Specifies the cluster instance ID data assigned to the OP2 on the second node with the following format: <i>SID COMPUTER_NAME HOST_NAME ORACLE_HOME</i> op2 OPSHP2 opshp2 c:\orant

During auto-discovery, the information in the registry yields a list of instances and nodes. Only instances that run on the local node are discovered. Once the instances and nodes are discovered, the LISTENER.ORA and TNSNAMES.ORA files are inspected.

LISTENER.ORA

Auto-discovery locates the listener and instance names for a node with the LISTENER.ORA file, located in `$ORACLE_HOME/network/admin` on UNIX platforms and `ORACLE_HOME\network\admin` on Windows NT on the discovered nodes.

Auto-discovery requires the following entries:

- The listener address must contain a TCP/IP address that specifies the HOST value as the host name of the discovered node. This ensures the listener actually resides on the node.

```
(description =  
  (address = (protocol = tcp)(host = opshp1-pc)(port = 1521))  
)
```

- For each listener that runs on the node, the `SID_LIST_listener_name` entry is searched for a description (`SID_DESC`) that contains the instance name (`SID_NAME`):

```
sid_list_listener =  
  (sid_list =  
    (sid_desc =  
      (sid_name = op1)  
    )  
  )
```

The LISTENER.ORA file created after installation typically contains the necessary configuration for auto-discovery. For further information, see "[Configuring Net8 for Nodes](#)" on page 4-12.

Note: On UNIX platforms, listeners and instances may also be specified in the *DB_NAME.CONF* file with the *inst_sid_list* and *lsnr_listener_name* parameters, as described in "[Parameter Descriptions](#)" on page B-5.

```
inst_sid_list = (op1, op2)
lsnr_listener_name = "listener_%m"
```

where %m expands to the node name.

TNSNAMES.ORA

The TNSNAMES.ORA file, located in *\$ORACLE_HOME/network/admin* on UNIX platforms and *ORACLE_HOME\network\admin* on Windows NT on the discovered nodes, is consulted by auto-discovery to determine names and address for the Oracle Parallel Server and instances on a node.

Auto-discovery requires the following entries:

- Each instance must have an entry in TNSNAMES.ORA

Shown below is instance OP1 defined:

```
op1 =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data =
      (service_name = op.us.oracle.com)
      (instance_name = op1))
```

- The Oracle Parallel Server should have an entry.

Shown below is database OP defined, as described in ["Creating An Entry in the TNSNAMES.ORA File for the Database and Enabling Client Load Balancing"](#) on page 4-25:

```
op =
  (description=
    (address=
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data=
      (service_name=op.us.oracle.com)))
```

- On Windows NT, each instance must have a startup entry named *SID_* startup that uses SID= in its CONNECT_DATA, as described in ["Additional Configuration Required for Windows NT in the TNSNAMES.ORA File"](#) on page 4-29.

```
op1_startup =
  (description =
    (address =
      (protocol = tcp)
      (host = opshp1)
      (port = 1521))
    (connect_data =
      (sid = op1))
```

For further information about creating a TNSNAMES.ORA on the managed nodes, see ["Step 2: Configure Net Service Names"](#) on page 4-20.

Auto-Discovery Results

Auto-discovery results in the creation of:

- Discovered nodes and services listed in SERVICES.ORA file in the \$ORACLE_HOME/network/agent directory on UNIX platforms and ORACLE_HOME\network\admin directory on Windows NT.

The SERVICES.ORA file should contain an OPS_DATABASE entry for the Oracle Parallel Server, which lists the node, database address, and name of the database. The example below show a database named OP.US.ORACLE.COM running on node OPSHP1. The database address comes from a OP.US.ORACLE.COM net service name in the TNSNAMES.ORA file.

```
op.us.oracle.com = (ops_database, opshp1, (description_
list=(description=(address=(protocol=tcp) (host=opshp1) (port=1521))
(connect_data=(sid=op1))) (description=(address=(protocol=tcp) (host=opshp2)
(port=1521)) (connect_data=(sid=op2))))), op.us.oracle.com)
```

The SERVICES.ORA file should also contain an OPS_INSTANCE entry for the instance that runs on the node. This entry identifies the:

- node name of the node
- address of instance obtained from TNSNAMES.ORA file
- Oracle Parallel Server name to which the instance belongs
- listener name

The example below shows the instance OP1 runs on node OPSHP1, and is listened for by LISTENER_OPSPHP1:

```
op1.us.oracle.com = (ops_instance, opshp1, (description=(address_
list=(address=(protocol=tcp) (host=opshp1) (port=1521))) (connect_
data=(sid=op1))), op.us.oracle.com, listener_opshp1)
```

- Errors logged inNMICONF.LOG in ORACLE_HOME/network/log directory on UNIX platforms and ORACLE_HOME\network\log directory on Windows NT.

The following message types should be ignored:

Warning: No Listener found for SID_db_name, db_name will be skipped.

Troubleshooting Auto-Discovery

If SERVICES.ORA contains an ORACLE_DATABASES entry instead of OPS_DATABASE and OPS_INSTANCE entries, discovery has failed. To resolve this:

1. Check NMICONF.LST in ORACLE_HOME/network/agent/config directory on UNIX platforms and ORACLE_HOME\network\agent\config directory on Windows NT. This file contains a list third-party auto-discovery scripts to run. It must contain at least the following entry:

```
confops.tcl
```

This entry is created during installation of the Oracle Parallel Server Option.

2. Check to see that the Oracle Parallel Server is defined correctly:

On UNIX:

- a. Verify oratab and the DB_NAME.CONF file are configured correctly.
- b. Run the following command to verify proper setup:

```
setenv ORACLE_PSRV db_name  
opsctl config -C
```

On Windows NT:

- a. Check the registry entries associated with the Oracle Parallel Server.
- b. Run the following command to verify proper setup:

```
opsctl config -ndb_name
```

OPSCTL displays the name of the node, instance and listener for the node. The example below shows a node named NODE1 running an instance named OP1 with a listener named LISTENER_NODE1.

```
node1 op1 listener_node1
```

3. Inspect LISTENER.ORA and TNSNAMES.ORA entries to ensure that the required entries are present.

Using Trace Files

This section discusses the following trace file subjects:

- [Background Thread Trace Files](#)
- [User Thread Trace Files](#)
- [Alert File](#)
- [Error Call Trace Stack](#)

Background Thread Trace Files

Oracle Parallel Server background threads use trace files to record occurrences and exceptions of database operations, as well as errors. These detailed trace logs are helpful to Oracle support to debug problems in your cluster configuration.

Background thread trace files are created regardless of whether or not the `BACKGROUND_DUMP_DEST` parameter is set in the `INITDB_NAME.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:

- `$ORACLE_BASE/admin/db_name/bdump` on UNIX platforms
- `ORACLE_BASE\admin\db_name\bdump` on Windows NT

Oracle8i 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`, such as:

- `SIDDBWR.TRC`
- `SIDSMON.TRC`

Oracle Parallel Server trace information is reported in the following trace files:

Trace File	Description
<i>SIDBSP0</i> .TRC	Trace file for BSP process. This trace files shows errors associated with BSP.
<i>SIDLCKN</i> .TRC	Trace file for the LCK <i>n</i> process. This trace file shows lock request for other background processes.
<i>SIDLMDN</i> .TRC	Trace file for the LMD <i>n</i> process. This trace file shows lock requests.
<i>SIDLMON</i> .TRC	Trace file for the LMON process. This trace file show status of cluster.
<i>SIDP00N</i> .TRC	Trace file for the parallel query slaves.

User Thread Trace Files

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 `ORAXXXXXX.TRC`, where `XXXXXX` is a 5-digit number indicating the process ID on UNIX or the Windows NT thread ID.

Alert File

The alert file, `SIDALRT.LOG`, contains important information about error messages and exceptions that occur during database operations. Each 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.

`SIDALRT.LOG` is found in the directory specified by the `BACKGROUND_DUMP_DEST` parameter in the `INITDB_NAME.ORA` initialization parameter file. If the `BACKGROUND_DUMP_DEST` parameter is not set, the `SIDALRT.LOG` file is generated in:

- `$ORACLE_BASE/admin/db_name/bdump` on UNIX platforms
- `ORACLE_BASE\admin\db_name\bdump` on Windows NT

Error Call Trace Stack

Oracle Worldwide Support may ask you to create an error call trace stack for a particular trace file. An error call trace stack provides program trace of specific background or user threads in the database.

To create an error call trace:

1. Obtain the Oracle process ID for the background processes:

```
sql> CONNECT internal/password
SELECT pid "Oracle Process Id",
       name
       from v$process, v$bgprocess
       where v$process.addr = v$bgprocess.paddr;
```

Output displayed looks like this:

```
Oracle Pro NAME
-----
2 PMON
3 LMON
4 LMD0
5 DBW0
6 LGWR
7 CKPT
8 SMON
9 RECO
10 SNP0
11 SNP1
13 LCK0
```

2. Dump the trace stack to the trace file. For example, to dump out the trace stack of LMON, enter:

- a. Set the Oracle process ID to LMON, which is 3 in this example:

```
sql> ORADEBUG setorapid 3
```

- b. Dump the error stack to *SIDLMON.TRC*:

```
sql> ORADEBUG dump errorstack 3
```

Contacting Oracle Worldwide Customer Support

If after reading this appendix, you still cannot resolve your problems, call Oracle Worldwide Customer Support to report the error. Please have the following information at hand:

- cluster hardware, for example, a two-node cluster of Dell PowerEdge 6100 servers
- operating system version
- all five digits in release number of Oracle RDBMS (for example, 8.1.5.0.0)
- all five digits in release number of Oracle Parallel Server Option
- on Windows NT, contents of HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\OSD key
- cluster OSD upgrades from vendor
- particular operation that failed, for example, database startup or query
- steps to reproduce the problem.

Severe Errors

If an ORA-600 error occurred, it will be printed to *SIDALRT.LOG* file. If an ORA-600 error or any other severe errors appear in the *SIDALRT.LOG* file, then provide all files in:

- `$ORACLE_BASE/admin/db_name/bdump` on UNIX platforms
- `ORACLE_BASE\admin\db_name\bdump` on Windows NT

Glossary

alert file

The alert file contains important information about error messages and exceptions that occur during database operations.

auto-discovery

When you execute the Discover Node command from the *Console*, the *Management Server* contacts the *Oracle Intelligent Agent* installed on that node to discover the Oracle services installed on the node. The Management Server then places the new information in the repository, and updates the hierarchical tree in the Navigator window of the Console, displaying a big-picture view of all nodes and their respective services.

BSP0 process

The BSP0 process manages out-going messages to requesting nodes for the Consistent Read (CR) Server.

cluster

A cluster is a set of instances, each typically running on different nodes, that coordinate with one another when accessing the shared database residing on disk.

Cluster Manager (CM)

Cluster Manager is an Operating System Dependent component that discovers and tracks the membership state of nodes by providing a common view of cluster membership across the cluster.

CM monitors process health, specifically of the Oracle8i release 8.1 database instance. The *LMON process*, a background process that monitors the health of the *Integrated Distributed Lock Manager (IDLM)*, registers and de-registers from CM.

Console

The *Oracle Enterprise Manager* Console gives you a central point of control for the Oracle environment through an intuitive graphical user interface (GUI) that provides powerful and robust system management.

fine-grain locks

A type of locking where PCM locks are dynamically allocated at block-access time. The resources for the lock are only allocated during the time the lock is needed and are released when the lock is released.

global database name

The global database name is the full name of the database which uniquely identifies it from any other database. The global database name is of the form "*database_name.database_domain*". For example:

OP.US.ORACLE.COM

The database name portion, OP, is a simple name you wish to call your database. The database domain portion, .US.ORACLE.COM, specifies the database domain in which the database is located, making the global database name unique. When possible, Oracle Corporation recommends that your database domain mirror the network domain.

The global database name is the default service name of database, as specified by the *SERVICE_NAMES* parameter in the *INITDB_NAME.ORA* file.

hashed locks

A type of locks where PCM locks are statically assigned to one or more blocks in the data files.

HOME_NAME

HOME_NAME represents the name of an Oracle home in an *Oracle service* on Windows NT.

INITSID.ORA

The *INITSID.ORA* file is an initialization file that contains parameters unique for an instance and points to *INITDBNAME.ORA* for database parameters.

INITDBNAME.ORA

The *INITDBNAME.ORA* file is an initialization file that contains database parameters.

Input/Output (IO)

Input/Output is an Operating System Dependent component that provides I/O to access shared disks.

instance

For an Oracle Parallel Server database, each node within the cluster has an instance of the running Oracle8i software referencing the database.

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. You can connect to any instance to access any information that resides within a parallel server database.

Each instance has unique *Oracle System Identifier (SID)*, *instance name*, *rollback segments*, and *thread ID*.

instance name

Instance name represents the name of the instance and is used to uniquely identify a specific instance when multiple instances share common services names. The instance name is identified by the INSTANCE_NAME parameter in the *INITSID.ORA* file.

The instance name is the same as the *Oracle System Identifier (SID)*.

Integrated Distributed Lock Manager (IDLM)

Oracle Parallel Server software that provides locking mechanisms to control allocation and modification of Oracle resources.

Inter-Process Communication (IPC)

Inter-Process Communication is an Operating System Dependent component that reliable transfer of messages between instances on different nodes.

LCKN process

The LCKN process manages the locks used by an instance and coordinates requests for those locks by other instances. Additional lock processes, LCK1 through LCK9, are available for systems that require exceptionally high throughput of instance lock requests. The single lock process per instance, LCK0, is usually sufficient for most systems.

listener

The listener is a separate process that resides on the server whose responsibility is to listen for incoming client connection requests and manage the traffic to the server.

The listener brokers the client request, handing off the request to the server. Every time a client (or server acting as a client) requests a network session with a server, a listener receives the actual request. If the client's information matches the listener's information, the listener grants a connection to the server.

LISTENER.ORA

A configuration file that describes one or more [listeners](#) on a server. This file must reside on each of the nodes.

LMDN process

The LMDN process handles remote lock requests (those which originate from another instance).

LMON process

The LMON process manages instance and processes deaths and associated recovery for the [Integrated Distributed Lock Manager \(IDLM\)](#).

Management Server

The [Oracle Enterprise Manager](#) Management Server provides centralized intelligence and distributed control between the [Console](#) and the managed nodes, and processes system management tasks sent by the Console and administers the distribution of these tasks across the enterprise. The Management Server stores all system data, application data, and information about the state of managed nodes in a repository, which is a set of tables stored in a database. High performance and scalability is ensured because the workload is automatically shared and balanced when there are multiple Management Servers.

multiple Oracle homes

The capability of having more than one Oracle home directory on a machine.

net service name

A net service name is able to access a service across the network by providing the network description information necessary to locate the service on the network.

Net8

Net8 is the foundation of Oracle's family of networking products, allowing services and their applications to reside on different computers and communicate as peer applications. The main function of Net8 is to establish network sessions and transfer data between a client machine and a server or between two servers. Once a network session is established, Net8 acts as a data courier for the client and the server.

node

A node is a machine where an instance resides.

OPSQL utility

Oracle Parallel Server Management uses the OPSQL utility (installed on each node) to manage instances. OPSQL gathers information about all the instances for *Oracle Enterprise Manager*. OPSQL serves as a single point of control between the Oracle *Oracle Intelligent Agent* and the nodes. Only one node's Oracle Intelligent Agent is used to communicate to OPSQL. OPSQL on that node then communicates to the other nodes through *Net8*.

Oracle Enterprise Manager

Oracle Enterprise Manager is a system management tool which provides an integrated solution for centrally managing your heterogeneous environment. Oracle Enterprise Manager combines a graphical *Console, Management Server, Oracle Intelligent Agent, repository database*, and tools to provide an integrated, comprehensive systems management platform for managing Oracle products.

A product family consists of system management tools designed to efficiently manage the complete Oracle environment.

Oracle Data Gatherer

The Oracle Data Gatherer collects performance statistics for the *Oracle Performance Manager*. The Oracle Data Gatherer must be installed on a node somewhere on the network.

Oracle Intelligent Agent

The Oracle Intelligent Agent is a process that runs on each of the nodes. It functions as the executor of jobs and events sent by the console via the *Management Server*. High availability is ensured since the agent can function regardless of the status of the *Console* or network connections.

Oracle Parallel Query

Parallel query divides the work of processing certain types of SQL statements among multiple query server processes.

Oracle Parallel Server

Oracle Parallel Server is an architecture that allows multiple instances to access a shared database of data files.

Oracle Parallel Server Option

The Oracle Parallel Server Option provides the necessary Oracle Parallel Server scripts, initialization files, and data files to make Oracle8i Enterprise Edition an Oracle Parallel Server database.

Oracle Parallel Server Management

Oracle Parallel Server Management is a comprehensive and integrated system management solution for the Oracle Parallel Server. Oracle Parallel Server Management allows you to manage multi-instance databases running in heterogeneous environments through an open client-server architecture through [Oracle Enterprise Manager](#).

In addition to managing parallel databases, Oracle Parallel Server Management allows you to schedule jobs, perform event management, monitor performance, and obtain statistics to tune parallel databases.

Oracle Performance Manager

Oracle Performance Manager is an add-on application for [Oracle Enterprise Manager](#) that offers a variety of tabular and graphic performance statistics for parallel servers. The statistics represent the aggregate performance for all instances running on an Oracle Parallel Server.

Oracle service

Oracle services are created and associated with Oracle products, such as the database or listener.

Oracle System Identifier (SID)

The Oracle System Identifier (SID) identifies a specific instance of the running Oracle8i software. For an Oracle Parallel Server database, each node within the cluster has an instance referencing the database.

The database name, specified by the DB_NAME parameter in the INITDB_NAME.ORA file, and unique *thread ID* make up each node's SID. The thread ID starts at 1 for the first instance in the cluster, and is incremented by 1 for the next instance, and so on.

Oracle8i Enterprise Edition

Oracle8i Enterprise Edition is an Object-Relational Database Management System (ORDBMS). It provides the applications and files to manage a database. All other Oracle Parallel Server components are layered on top of Oracle8i Enterprise Edition.

Operating System Dependent layer

A software layer that consists of several software components developed by vendors. The OSD layer maps the key OS/cluster-ware services required for proper operation of Oracle Parallel Server.

OPCTL

A utility to start and stop instances from either one node or from *Oracle Enterprise Manager*.

OPS Communication Daemon (OPSD)

The OPS Communication Daemon (OPSD) receives requests from the *OPCTL utility* to execute administrative job tasks, such as startup or shutdown. The command is executed locally on each node, and the results are sent back to OPCTL. OPSD is installed on the nodes. OPSD is only implemented on UNIX platforms.

Parallel Cache Management (PCM)

Oracle Parallel Server software that provides instance locks that cover one or more data blocks (table or index blocks) in the buffer cache. PCM locks ensure that each shared buffer cache in a node remains consistent with the shared buffer caches in other nodes with minimal use of the *Integrated Distributed Lock Manager (IDLM)*.

parallel coordinator

A process that dispatches the execution of a SQL statement to several *parallel server processes* and coordinates the results from all of the slave processes to send the results back to the user.

parallel server process

Processes that performs the requested action of the *parallel coordinator*.

preferred credentials

Each *Oracle Enterprise Manager* administrator can set up specific user names, passwords, and roles for nodes, listeners, databases, and other services that you administer in the network.

After these credentials are set up, you log in once to start the *Console* and are then automatically logged in as needed to the nodes. All login credentials are encrypted in the *repository database*.

repository database

A repository database is a set of tables in an Oracle database which stores data required by *Oracle Enterprise Manager*. This database is separate from the database on the nodes.

rollback segments

Records the old values of data that was changed by each transaction (whether committed or not committed). Rollback segments are used to provide read consistency, to roll back transactions, and to recover the database. Each node typically has two rollback segments, which are identified with a naming convention of *RBS**thread_id**rollback_number* by the `ROLLBACK_SEGMENTS` parameter in the *INITSID.ORA* file.

service name

The service name identifies the database. By default, the Oracle Universal Installer and the Oracle Database Configuration Assistant creates a service name that includes the entire *global database name*, a name comprised of the database name (`DB_NAME`) and domain name (`DB_DOMAIN`), entered during installation or database creation.

The service name is identified by the `SERVICE_NAMES` parameter in the *INITDBNAME.ORA* file.

SID

SID is an abbreviation for [Oracle System Identifier \(SID\)](#).

SETLINKS utility

SETLINKS is a utility to map symbolic links to raw devices for Windows NT.

Startup (START)

Startup is an Operating System Dependent component that provides one-time configuration to startup functionality.

SYSDBA

SYSDBA is a special database administration role that contains all system privileges with the ADMIN OPTION, and the [SYSOPER](#) system privilege. SYSDBA also permits CREATE DATABASE and time-based recovery.

SYSOPER

SYSOPER is 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](#).

tablespace

A logical portion of an Oracle database used to allocate storage for table and index data. Each tablespace corresponds to one or more physical data files. Every Oracle database has a tablespace called SYSTEM and may have additional tablespaces. A tablespace is used to group related logical structures together. For example, tablespaces commonly group all of an application's objects to simplify certain administrative operations.

thread ID

The number of the redo thread that is to be used by the instance. Any available redo thread number can be used, but an instance cannot use the same thread number as another instance. Also, an instance cannot start when its redo thread is disabled. An instance cannot mount a database if the thread is used by another instance or if the thread is disabled.

The thread starts at 1 node for the first instance in the cluster, and is incremented by 1 for the next instance, and so on..

Threads are depicted by the THREAD parameter in the *INITSID.ORA* file.

When redo log files are generated, they include the thread ID, allowing you to easily identify a particular node's log files.

TNSNAMES.ORA

A file that contains *net service names*. This file is needed on clients, nodes, the *Console*, and the *Oracle Performance Manager* machine.

trace file

Each server and background process can write to an associated trace file. When an internal error is detected by a process, 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.

Index

A

abort mode, warning notice, 6-22
administering instances, 6-1 to 6-39
 with initialization files, 6-2
 with Oracle Parallel Server Management, 6-13
alert log, C-12
ALTER DATABASE ADD LOGFILE, 8-2
architecture
 OPSQL utility on UNIX, 5-4
 OPSQL utility on Windows NT, 5-6
 Optimal Flexible Architecture (OFA), 1-2
 Oracle Enterprise Manager, 5-8
 Oracle Parallel Server, 1-2
 Oracle Parallel Server Management, 5-2
auto-discovery
 DB_NAME.CONF file, C-4
 errors, C-3
 LISTENER.ORA file, C-6
 NMICONF.LOG file, C-9
 NMICONF.LST file, C-10
 on UNIX, C-4
 on Windows NT, C-5
 performing, 5-18
 PM\DB_NAME registry key, C-5
 SERVICES.ORA file, C-9
 SID_LIST_listener_name parameter, C-6
 TNSNAMES.ORA file, C-7
 troubleshooting, C-10
 understanding, C-3

B

BACKGROUND_DUMP_DEST parameter, C-11, C-12
backslash, defined, xvi
BACKUP parameter, 4-44
BSP0 process, 1-10

C

charts
 OPS Overview, 7-19
 Oracle Performance Manager, 7-1 to 7-19
 Parallel Server Active Users, 7-17
 Parallel Server Active Users by Instance, 7-18
 Parallel Server Block Ping, 7-6
 Parallel Server Block Ping Tablespace
 Instance, 7-7
 Parallel Server Block Ping Tablespace
 Object, 7-7
 Parallel Server Data Block Ping, 7-7
 Parallel Server Data Block Ping by Instance, 7-9
 Parallel Server File I/O Rate, 7-10
 Parallel Server File I/O Rate by Instance, 7-12
 Parallel Server File I/O Rate by Object, 7-11
 Parallel Server Lock Activity, 7-13
 Parallel Server Sessions, 7-14
 Parallel Server Users Logged On, 7-15
 Parallel Server Users Logged On by
 Instance, 7-16
CHECKPOINT_PROCESS parameter, 6-10
Choose Start >, defined, xvi
Client, 4-25
client load balancing, 4-25
cluster management, 1-7

- Cluster Manager (CM)
 - defined, 1-4
 - See CM, 1-6
- cluster, defined, 1-9
- CM
 - cluster management, 1-7
 - defined, 1-4
 - DefinedNodes registry value, 4-6
 - described, 1-6
 - determining groups available, 1-7
 - determining instances available, 1-7
 - Group Membership Service (GMS), 1-7
 - interacting with IDLM, 1-6, 1-16
 - node monitoring, 1-6
- coexistence, 3-8
- Configuration Assistant, 5-13
- configuration tasks
 - adding nodes to a cluster, 8-6
 - configuring the listener, 4-13
 - configuring the listener for failover, 4-36
 - creating net service names, 4-20
 - creating the database, 3-6
 - database creation, 4-3
 - establishing client load balancing, 4-25
 - establishing connect-time failover, 4-35
 - establishing transparent application failover, 4-43
 - moving from single instance to parallel server, 8-2 to 8-5
 - Net8 on clients, 4-34
 - Net8 on Console, 5-11
 - Net8 on nodes, 4-12
 - Net8 on Oracle Performance Manager machine, 5-29
 - OPSQL utility on UNIX, B-1 to B-12
 - Oracle Enterprise Manager, 5-11 to 5-28
 - Oracle Intelligent Agents on UNIX, B-11 to B-12
 - Oracle Parallel Server, 4-1 to 4-45
 - Oracle Parallel Server Management, 5-1 to 5-37
 - Oracle Performance Manager, 5-29 to 5-37
 - testing the network, 4-45
 - with Oracle Database Configuration Assistant, 4-4
 - connect descriptor, defined, 4-20
 - connecting to the database, 4-45
 - connecting to the repository, 5-31, 5-36
 - connect-time failover, 4-35
 - Try each address, in order, until one succeeds option, 4-41
 - Try each address, randomly, until one succeeds option, 4-41
 - with transparent application failover, 4-43
 - Consistent Read Server, 1-10
 - Console
 - defined, 2-3, 5-2
 - Event Management window, 5-2
 - Group window, 5-2
 - Job window, 5-2
 - Navigator window, 5-2
 - Navigator window with Oracle Parallel Server objects, 6-14
 - right-mouse menu, 6-17
 - Connect, 6-17
 - Disconnect, 6-17
 - Edit, 6-17, 6-27
 - Related Tools, 6-17
 - Remove, 6-17
 - Results, 6-17, 6-24
 - Shutdown, 6-17, 6-21
 - Startup, 6-17, 6-18
 - scheduling jobs, 6-31
 - setup, 5-8
 - starting, 5-15, 6-14
 - starting Oracle Parallel Server database, 6-18
 - stopping Oracle Parallel Server database, 6-21
 - viewing Oracle Parallel Server status, 6-27
 - viewing shutdown results, 6-24
 - viewing startup results, 6-24
 - control files, 1-10
 - control files raw devices, 2-7
 - CONTROL_FILES parameter, 6-11
 - CPU_COUNT parameter, 6-11
 - CR Server, 1-10
 - Create Job property sheet, 6-32 to 6-39
 - General tab, 6-33
 - Parameters tab, 6-37
 - Tasks tab, 6-35

- creating
 - database, 3-5
 - DB_NAME.CONF* file, B-2
 - net service name for each database, 4-25
 - net service names, 4-20
 - net service names for each instance, 4-20
 - net service names for Windows NT, 4-29
 - network configuration files, 3-6, 4-12, 5-11, 5-29
 - preferred credentials, 5-21
 - raw devices, 8-3
 - raw devices on UNIX, 2-7
 - raw devices on Windows NT, 2-9
 - repository, 5-13
- Custom installation, 3-6
- Custom installation type, 3-2, 4-2

D

- data files, 1-10, 1-11, 1-14
- database
 - creating, 3-5, 3-6, 4-3
 - creating a job on, 6-31
 - disconnecting, 6-22
 - displaying objects in Console, 6-14
 - mounting, 6-19
- database creation, 3-5, 3-6, 4-3
- database creation failure, C-2
- database, defined, 1-9
- \database directory, A-3
- DB, 2-9, 4-19
- DB_BLOCK_SIZE* parameter, 6-11
- DB_DOMAIN* parameter, 6-7, 6-11
- DB_FILES* parameter, 6-11
- DB_NAME* parameter, 1-11, 6-3, 6-7, 6-11
- DB_NAME.CONF* file
 - auto-discovery, C-4
 - creating, B-2
 - environ* parameter, B-6
 - examples, B-7
 - inst_init_ora* parameter, B-5, B-8
 - inst_oracle_sid* parameter, B-5, B-7
 - inst_sid_list* parameter, C-7
 - listener configuration, B-9
 - lsnr_listener_name* parameter, 4-19, B-5, B-10, C-7
 - node#tns_admin* parameter, B-5
 - node#inst_init_ora* parameter, B-5

- node_list* parameter, 5-12, B-5, C-4
- oracle_home* parameter, B-5
- overview, B-2
- parameter expansion, B-3
- parameter syntax, B-3
- parameters, B-5
- parameters node overrides, B-7
- sample, B-11
 - tns_admin* parameter, B-5, B-9
- DB_NAME.TBL* file, 2-9
- DBA locking, 1-18
- DBCA, 2-8, 2-13, 4-3
- DBCA_RAW_CONFIG environment variable, 2-8, 2-13, 4-3
- /dbs directory, A-2
- DBWR process, 1-10
- DefinedNodes registry value, 4-6
- DELAYED_LOGGING_BLOCK_CLEANOUPS parameter, 6-10
- directory structure, A-1 to A-3
 - UNIX, A-2
 - Windows NT, A-3
- discovering nodes, 5-18
- discovery
 - See auto-discovery
- Distributed Lock Manager, described, 1-10
- DLM, described, 1-10
- DML_LOCKS* parameter, 6-10, 6-11
- DR tablespace raw device, 2-6

E

- Edit Parallel Server dialog box, 6-27
- error call trace stack, C-13
- EXP, 8-3
- export data from old database, 8-3
- Export utility, 8-3

F

FAILOVER, 4-35, 4-41, 4-43
failover
 connect-time, 4-35
 transparent application failover, 4-35
FAILOVER parameter, 4-35, 4-41
FAILOVER_MODE parameter, 4-35, 4-43
fine-grain locks, 1-18
 1 block-to-1 block ratio, 1-18
 fixed, 1-18
 releasable, 1-18
 when to use, 1-19
FORCE mode, warning notice, 6-19
forward slash, defined, xvi

G

GC_FILES_TO_LOCKS parameter, 1-17, 6-9, 6-11
GC_RELEASABLE_LOCKS parameter, 1-18, 6-9
GC_ROLLBACK_LOCKS parameter, 6-9, 6-11
General tab, from Create Job property sheet, 6-33
Global Constant parameters list, 6-9
global database name, 1-9, 3-4, 4-8
global VS view tables, 5-1, 7-2
GMS
 described, 1-7
 release 8.0 functionality, 1-7
GUI, 5-15
GV\$FILESTAT view, 7-10, 7-11, 7-12
GV\$LICENSE view, 7-15, 7-16, 7-17, 7-18
GV\$LOCK_ACTIVITY view, 7-13
GV\$PING view, 7-6, 7-7, 7-9
GV\$PROCESS view, 7-14
GV\$SESSION view, 7-14

H

hardware requirements, 2-2
hashed locks, 1-18
 fixed, 1-18
 releasable, 1-18
 when to use, 1-19
HOSTS.EQUIV file, 2-14

I

IDLM
 block access synchronization, 1-16
 deadlock detection, 1-16
 defined, 1-15
 described, 1-15
 interacting with CM, 1-6, 1-16
 LMDN process, 1-15
 LMON process, 1-15
 modification synchronization, 1-16
IFILE parameter, 6-4, 6-5
IMP, 8-5
import data, 8-5
Import utility, 8-5
INDX1 tablespace raw device, 2-6
INITDB_NAME.ORA file, 1-11, 6-2
 BACKGROUND_DUMP_DEST
 parameter, C-11, C-12
 CHECKPOINT_PROCESS parameter, 6-10
 CONTROL_FILES parameter, 6-11
 CPU_COUNT parameter, 6-11
 DB_BLOCK_SIZE parameter, 6-11
 DB_DOMAIN parameter, 6-7, 6-11
 DB_FILES parameter, 6-11
 DB_NAME parameter, 1-11, 6-3, 6-7, 6-11
 DELAYED_LOGGING_BLOCK_
 CLEANOUTS parameter, 6-10
 DML_LOCKS parameter, 6-10, 6-11
 GC_FILES_TO_LOCKS parameter, 1-17, 6-9,
 6-11
 GC_RELEASABLE_LOCKS parameter, 1-18, 6-9
 GC_ROLLBACK_LOCKS parameter, 6-9, 6-11
 Global Constant (GC) parameters, 6-8
 identical parameters, 6-11
 INSTANCE_NUMBER parameter, 6-10
 LM_LOCKS parameter, 6-11, 6-12
 LM_PROCS parameter, 6-11, 6-12
 LM_RESS parameter, 6-11, 6-12
 LOCAL_LISTENER parameter, 3-6, 4-17, 4-37
 LOG_ARCHIVE_FORMAT parameter, 6-10
 MAX_COMMIT_PROPAGATION_DELAY
 parameter, 6-10, 6-11
 NLS_* parameters, 6-10

PARALLEL_DEFAULT_MAX_INSTANCES
 parameter, 6-11
 PARALLEL_DEFAULT_MAX_SCANS
 parameter, 6-11
 PARALLEL_SERVER parameter, 6-10
 parameter notes, 6-10
 PROCESSES parameter, 6-10
 RECOVERY_PARALLELISM parameter, 6-10
 ROLLBACK_SEGMENTS parameter, 6-10, 6-11
 ROW_LOCKING parameter, 6-11
 SERVICE_NAMES parameter, 6-7
 THREAD parameter, 6-10
 USER_DUMP_DEST parameter, C-12
 initialization files, 6-1 to 6-39
 definition, 6-2
 editing, 6-12
 for instances, 6-2
 identical parameters, 6-11
 INITDB_NAME.ORA, 6-2
 INITSID.ORA, 6-2
 location, 6-12
 initialization parameters
 descriptions, 6-12
 editing, 6-12
 Global Constant (GC) parameters, 6-8
 identical, 6-11
 setting, 6-8
 INITSID.ORA file, 1-11, 6-2, 6-4
 IFILE parameter, 6-4, 6-5
 INSTANCE_NAME parameter, 1-11, 4-24, 6-4
 REMOTE_LOGIN_PASSWORDFILE
 parameter, 6-4
 ROLLBACK_SEGMENTS parameter, 1-12
 THREAD parameter, 1-12, 6-4
 Input/Output (IO)
 defined, 1-4
 described, 1-7
 on Windows NT, 1-4
 inst_init_ora parameter, B-5, B-8
 inst_oracle_sid parameter, B-5, B-7
 inst_sid_list parameter, C-7
 installation, 3-7
 Custom install type, 3-2, 4-2
 Custom procedure, 3-6
 database creation, 3-5, 3-6
 directory structure, A-1 to A-3
 INITDB_NAME.ORA, 3-5, 4-8
 INITSID.ORA, 3-5, 4-8
 Minimal install type, 3-2
 Net8 Configuration Assistant, 3-6
 network configuration, 3-6
 Operating System Dependent layer, 2-13, 8-3
 Oracle Data Gatherer, 5-28
 Oracle Database Configuration Assistant, 3-5,
 3-6
 Oracle Diagnostics Pack, 5-28
 Oracle Enterprise Manager, 5-10
 Oracle Enterprise Manager requirements, 2-3
 Oracle Parallel Server Option, 3-3, 8-4
 Oracle Performance Manager, 5-28
 Oracle Universal Installer, 3-3
 Oracle8i Enterprise Edition, 3-3, 8-4
 pre-installation, 2-1 to 2-15
 procedure, 3-3
 repository database, 5-10
 software requirements, 2-2
 Typical install type, 3-2, 4-2
 Typical procedure, 3-5
 verifying raw devices, 3-5, 3-6
 installation requirements, 2-2 to 2-5
 disk space, 2-5
 hardware, 2-2
 hardware requirements, 2-2
 Oracle Enterprise Manager, 2-3
 shared disk partitions, 2-5
 software, 2-2
 installation type
 Custom, 3-2, 4-2
 Minimal, 3-2
 Typical, 3-2, 4-2
 instance name, 1-11
 INSTANCE_NAME parameter, 1-11, 4-24, 4-29, 6-4
 INSTANCE_NUMBER parameter, 6-10

instances

- administering, 6-1 to 6-39
- BSPO process, 1-10
- control files, 1-10
- data files, 1-10, 1-11, 1-14
- DBWR process, 1-10
- defined, 1-9
- Destination Type tasks for, 6-36
- examples of parameters in
 - DB_NAME.CONF* file, B-7
- identical parameters, 6-11
- initialization files, 6-2
- INITSID.ORA, 1-11
- instance name, 1-11
- INSTANCE_NAME parameter, 1-11, 4-24
- Integrated Distributed Lock Manager (IDLM), 1-10
- issues, 6-8
- LCKN process, 1-10
- LGWR process, 1-10
- LMD0 process, 1-10
- LMON process, 1-10
- management with Oracle Parallel Server Management, 6-13
- Oracle System Identifier (SID), 1-11
- ORACLE_SID environment variable, 1-11
- ORACLE_SID registry entry, 1-11
- PMON process, 1-10
- rollback segments, 1-12
- ROLLBACK_SEGMENTS parameter, 1-12
- setting initialization parameters, 6-8
- SID, 1-11, 6-3
- SID Prefix, 4-8
- SID prefix, 3-4
- SMON process, 1-10
- thread ID, 1-12
- THREAD parameter, 1-12
- verifying, 4-11
- viewing startup and shutdown results, 6-24, 6-27

Integrated Distributed Lock Manager (IDLM)

- defined, 1-15
- described, 1-15
- See IDLM

Inter-Process Communication (IPC)

- defined, 1-4
- described, 1-7

Introducing, 1-1

italic, defined, xvi

J

jobs

- choosing tasks for, 6-35
- creating for Oracle Parallel Server, 6-31
- setting up rights for running, 6-38
- specifying information for, 6-33
- specifying parameters for OPSM tasks, 6-37

L

LCKN process, 1-10

LGWR process, 1-10

listener

- configuring, 4-13
- configuring for failover, 4-36
- configuring *lsnr_listener_name* parameter, B-10
- configuring *tns_admin* parameter, B-9
- starting, 4-18, 4-39
- stopping, 4-16, 4-37

LISTENER.ORA file, 4-12, 5-7, B-9

- configuring, 4-13
- configuring for failover, 4-36
- default configuration, 4-14
- how used for auto-discovery, C-6
- SID_DESC parameter, C-6
- SID_LIST_listener_name parameter, 4-12, 4-37, C-6
- SID_NAME parameter, C-6

LM_LOCKS parameter, 6-11, 6-12

LM_PROCS parameter, 6-11, 6-12

LM_RESS parameter, 6-11, 6-12

LMD0 process, 1-10

LMDN process, 1-15

LMON process, 1-10, 1-15

LOAD, 4-25, 4-41

LOAD_BALANCE parameter, 4-25, 4-41

LOCAL_LISTENER parameter, 3-6, 4-17, 4-37

Lock Manager, 4-4

LOG_ARCHIVE_FORMAT parameter, 6-10
lowercase courier, defined, xvi
lsnr_listener_name parameter, 4-19, B-5, B-10, C-7
LSNRCTL control utility
 DBSNMP START command, 5-13, B-12
 START command, 4-18, 4-39
 STOP command, 4-16, 4-37

M

managed targets, defined, 2-4
Management Server
 OMSCONFIG.PROPERTIES file, 5-14
 starting, 5-14, 6-14
Management Server, defined, 2-3, 5-2
MAX_COMMIT_PROPAGATION_DELAY
 parameter, 6-10, 6-11
METHOD parameter, 4-45
migrating, Oracle Parallel Server issues, 3-8
Minimal installation type, 3-2
multiple Oracle homes
 UNIX, 3-9
 Windows NT, 3-9

N

NAMES_DIRECTORY_PATH parameter, 4-32
Navigator window
 Oracle Parallel Server folder, 6-14
 Oracle Parallel Server objects, 6-14
 Parallel Server Instances folder, 6-16
 right-mouse menu, 6-17, 6-24
 Connect, 6-17
 Disconnect, 6-17
 Edit, 6-17, 6-27
 Related Tools, 6-17
 Remove, 6-17
 Results, 6-17
 Shutdown, 6-17, 6-21
 Startup, 6-17, 6-18
 starting Oracle Parallel Server database, 6-18
 stopping Oracle Parallel Server database, 6-21
 viewing Oracle Parallel Server status, 6-27
 viewing shutdown results, 6-24
 viewing startup results, 6-24
net service names

 configuring, 4-20
 described, 4-12
Net8 Assistant
 configuring net service names for each
 instance, 4-20
 configuring net service names for the
 database, 4-25
 configuring net service names for Windows
 NT, 4-29
 configuring the listener, 4-16
 configuring the listener for failover, 4-37
 enabling client load balancing, 4-25
Net8 Configuration Assistant
 automatically running as a part of
 installation, 3-6
network configuration, 3-6, C-6, C-7
 additional steps for Windows NT, 4-29
 configuring connect-time failover, 4-35
 configuring net service names for Windows
 NT, 4-29
 configuring the listener, 4-13
 configuring the listener for failover, 4-36
 configuring transparent application
 failover, 4-43
 creating net service names, 4-20
 enabling client load balancing, 4-25
 on clients, 4-34
 on Console, 5-11
 on nodes, 4-12
 Oracle Performance Manager machine, 5-29
 testing components, 4-45
 transparent application failover, 4-35
network configuration file
 LISTENER.ORA, 4-13, 4-36
 LISTENER.ORA file
 how used for auto-discovery, C-6
 SQLNET.ORA, 4-32
 TNSNAMES.ORA file
 how used for auto-discovery, C-7

- network configuration files
 - LISTENER.ORA, 4-12, 5-7, B-9
 - on clients, 4-34
 - on Console, 5-11
 - on nodes, 4-12
 - on Oracle Performance Manager machine, 5-29
 - SQLNET.ORA, 4-13
 - testing, 4-45
 - TNSNAMES.ORA, 4-13, 4-20, 5-7, 5-11, 5-29
- NLS_* parameters, 6-10
- NMICONF, C-9, C-10
- NMICONF.LOG file, C-9
- NMICONF.LST file, C-10
- node, 5-12, C-4
- node monitoring, 1-6
- node#:*environ parameter, B-6
- node#:*inst_init_ora parameter, B-5
- node#:*tns_admin parameter, B-5
- node, defined, 1-9
- node_list parameter, 5-12, B-5, C-4
- non-shared locations, B-11

O

- OEM_REPOSITORY tablespace raw device, 2-6
- oemapp console command, 5-15
- oemctrl start oms command, 5-14
- OLTP, 1-21
- OMSCONFIG.PROPERTIES file, 5-14
- Operating System Dependent layer
 - Cluster Manager (CM), 1-6
 - components, 1-4
 - defined, 1-3
 - described, 1-4
 - Input/Output (IO), 1-7
 - installing, 2-13, 8-3
 - Inter-Process Communication (IPC), 1-7
 - Startup (START), 1-8

- OPS Communication Daemon (OPSD),
 - defined, 5-4
 - /ops directory, A-2, C-4
 - \ops directory, A-3
 - \ops\dbca directory, 4-3
 - \ops_preinstall directory, 2-13
 - OPS Overview chart, 7-19
 - OPS_DATABASE entry, C-9
 - OPS_DATABASE entry in SERVICES.ORA file, C-9
 - OPS_INSTANCE entry, C-9
 - OPS_INSTANCE entry in SERVICES.ORA file, C-9
 - OPSQL utility, 5-4
 - architecture on UNIX, 5-4
 - architecture on Windows NT, 5-6
 - configuring for UNIX, 5-12
 - configuring on UNIX, B-1 to B-12
 - DB_NAME.CONF file, B-2
 - OPSD, defined, 5-4
 - /opsm directory, A-2
 - OP.TBL, 4-3
 - Optimal Flexible Architecture (OFA), 1-2
 - ORA-600, C-14
 - Oracle, 2-4
 - Oracle Data Gatherer
 - defined, 5-3
 - starting, 5-30
 - Oracle Database Configuration Assistant
 - automatically running as a part of
 - installation, 3-5, 3-6
 - creating a database, 4-4
 - creating a database with, 4-4
 - error message, 4-6
 - failing on Windows NT, C-2
 - initialization files, 1-11
 - minimum size, 2-6
 - running after installation, 3-7
 - troubleshooting, 4-4, 4-6

- Oracle Diagnostics Pack
 - described, 2-4
 - Oracle Performance Manager, 2-4
 - installing, 5-28
- Oracle Enterprise Manager
 - architecture, 5-8
 - auto-discovery, 5-18, C-3
 - components, 2-3
 - Configuration Assistant, 5-13
 - configuring, 5-11 to 5-28
 - configuring Net8 on Console, 5-11
 - configuring Net8 on Oracle Performance Manager machine, 5-29
 - configuring OPSCCTL, 5-12
 - configuring repository, 5-13
 - Console, 2-3, 5-2, 5-15
 - discovering nodes, 5-18
 - hardware requirements, 2-3
 - installation, 5-10
 - installation requirements, 2-3
 - installing repository database, 5-10
 - managed targets, 2-4
 - Management Server, 2-3, 5-2, 5-14
 - NMICONF.LOG file, C-9
 - NMICONF.LST file, C-10
 - OPS Communication Daemon (OPSD), 5-4
 - OPSCCTL utility, 5-4
 - Oracle Data Gatherer, 5-3
 - Oracle Diagnostics Pack, 2-4
 - Oracle Intelligent Agents, 5-3
 - Oracle Parallel Server Management, 1-3
 - Oracle Performance Manager, 5-3
 - repository database, 2-4, 5-3
 - scheduling jobs, 6-31
 - SERVICES.ORA file, C-9
 - setting up preferred credentials, 5-21
 - setup scenarios, 5-8
 - SID_LIST_listener_name parameter, 4-12, 4-37, C-6
 - software requirements, 2-3
 - solving auto-discovery failures, C-3
 - specifying preferred credentials, 5-21
 - starting, 5-13
 - starting Oracle Data Gatherer, 5-30
 - starting Oracle Intelligent Agents, 5-13, 6-14
 - starting Oracle Intelligent Agents, B-12
 - starting Oracle Parallel Server database, 6-18
 - starting Oracle Performance Manager, 7-4
 - starting the Console, 5-15, 6-14
 - starting the Management Server, 5-14, 6-14
 - stopping Oracle Parallel Server database, 6-21
 - super administrator account, 5-15
 - SYSDBA privilege, 5-28
 - SYSOPER privilege, 5-28
 - understanding auto-discovery failures, C-3
 - User Manager, 5-22
 - version 2 changes, 5-11
 - viewing Oracle Parallel Server status, 6-27
 - viewing shutdown results, 6-24
 - viewing startup results, 6-24
- Oracle home
 - non-shared on UNIX, B-12
 - shared on UNIX, B-12
- Oracle home directory, defined, xvii
- Oracle Intelligent Agents
 - defined, 5-3
 - setup and location on UNIX, B-11 to B-12
 - starting, 5-13, 6-14, B-12
- Oracle Parallel Query
 - defined, 1-15
 - described, 1-20
 - issuing SQL statements, 1-22
- Oracle Parallel Server
 - architecture, 1-2
 - data files, 1-11, 1-14
 - database creation, 3-5, 3-6, 4-3
 - database files, 1-14
 - Destination Type tasks, 6-36
 - folder in Console Navigator window, 6-14
 - hardware requirements, 2-2
 - INITDB_NAME.ORA file, 1-11
 - installation requirements, 2-2
 - installing, 3-1
 - instances, 1-9
 - Integrated Distributed Lock Manager (IDLM), 1-15
 - Operating System Dependent layer, 1-3
 - Oracle Parallel Query, 1-15, 1-20
 - Oracle Parallel Server Management, 1-3
 - Oracle Parallel Server Option, 1-3

- Oracle8i Enterprise Edition, 1-3
 - overview, 1-1 to 1-22
- Parallel Cache Management (PCM), 1-15
 - pre-installation, 2-1 to 2-15
 - raw device setup, 2-6 to 2-12
 - software requirements, 2-2
- Oracle Parallel Server database
 - creation, 4-4, 8-4
 - shutdown, 6-24
 - starting, 4-9, 8-5
- Oracle Parallel Server Management
 - administration of instances, 6-13
 - architecture, 5-2
 - configuring, 5-1 to 5-37
 - defined, 1-3
 - LISTENER.ORA file, 5-7
 - node requirements, 5-7
 - OPS Communication Daemon (OPSD), 5-4
 - OPSCCTL utility, 5-4
 - Oracle Enterprise Manager, 1-3, 5-2
 - Oracle Performance Manager, 5-28
 - requirements, 5-7
 - SIDs, 5-7
 - TNSNAMES.ORA file, 5-7
- Oracle Parallel Server Option
 - defined, 1-3
 - installing, 3-3, 8-4
- Oracle Performance Manager
 - accessing Oracle Parallel Server charts, 5-34
 - charts, 7-1 to 7-19
 - configuring, 5-29 to 5-37
 - Databases folder, 5-36
 - defined, 2-4, 5-3
 - displaying charts, 7-4
 - installing, 5-28
 - installing Oracle Data Gatherer, 5-28
 - installing Oracle Diagnostics Pack, 5-28
 - Oracle Diagnostics Pack, 2-4
 - overview, 7-2
 - Parallel Server Instance folder, 5-34
 - starting, 5-31, 7-4
- Oracle System Identifier (SID), defined, 1-11
- Oracle Universal Installer
 - using, 3-3
- ORACLE_HOME environment variable, B-6
- oracle_home parameter, B-5
- ORACLE_HOME, defined, xvii
- ORACLE_PSRV environment variable, B-3, B-6
- ORACLE_SID environment variable, 1-11
- ORACLE_SID registry entry, 1-11
- Oracle8i Enterprise Edition
 - defined, 1-3
 - installing, 3-3, 8-4
- OracleHOME_NAMEAgent service, 5-13
- OracleHOME_NAMEDataGatherer service, 5-30
- OracleHOME_NAMEManagementServer
 - service, 5-14
- OracleServiceSID registry key
 - registry
 - OracleServiceSID key, C-2
- OracleServiceSID service, 4-9
- ORAXXXXX.TRC file, C-12
- Output tab
 - from Parallel Server Operation Results dialog
 - box, 6-26

P

- Parallel Cache Management (PCM)
 - defined, 1-15
 - described, 1-17
 - See PCM
- parallel coordinator, defined, 1-20
- PARALLEL option, 4-10
- Parallel Server Active Users by Instance chart, 7-18
- Parallel Server Active Users chart, 7-17
- Parallel Server Block Ping chart, 7-6
- Parallel Server Block Ping Tablespace Instance
 - chart, 7-7
- Parallel Server Block Ping Tablespace Object
 - chart, 7-7
- Parallel Server Data Block Ping by Instance
 - chart, 7-9
- Parallel Server Data Block Ping chart, 7-7
- Parallel Server File I/O Rate by Instance
 - chart, 7-12
- Parallel Server File I/O Rate by Object chart, 7-11
- Parallel Server File I/O Rate chart, 7-10
- Parallel Server Instances folder
 - in Console Navigator window, 6-16

- Parallel Server Lock Activity chart, 7-13
- Parallel Server Operation Results dialog box, 6-24
- parallel server process, defined, 1-20
- Parallel Server Sessions chart, 7-14
- Parallel Server Shutdown dialog box, 6-21
- Parallel Server Startup dialog box, 6-18
- Parallel Server Users Logged On by Instance chart, 7-16
- Parallel Server Users Logged On chart, 7-15
- PARALLEL_DEFAULT_MAX_INSTANCES parameter, 6-11
- PARALLEL_DEFAULT_MAX_SCANS parameter, 6-11
- PARALLEL_SERVER parameter, 6-10
- Parameters tab
 - from Create Job property sheet, 6-37
- PCM
 - defined, 1-15
 - described, 1-17
 - fine-grain lock usage, 1-19
 - fine-grain locks, 1-18
 - GC_FILES_TO_LOCKS parameter, 1-17
 - GC_RELEASABLE_LOCKS parameter, 1-18
 - hashed lock usage, 1-19
 - hashed locks, 1-18
- PM*DB_NAME* registry key, C-2
- PMON process, 1-10
- preferred credentials, 5-21
 - creating an operating system account, 5-22
 - database, 5-21
 - nodes, 5-21
 - SYSDBA or SYSOPER for database, 5-21
- preferred user credentials, described, 5-21
- pre-installation, 2-1 to 2-15
 - DBCA_RAW_CONFIG, 2-8, 2-13, 4-3
 - oracle account activities, 2-15
 - root user activities, 2-14
- PROCESSES parameter, 6-10

R

- raw devices
 - control files, 2-7
 - creating, 2-7, 8-3
 - creating as root user, 2-7
 - creating on UNIX, 2-7
 - creating on Windows NT, 2-9
 - DB_NAME.TBL* file, 2-9
 - DR tablespace, 2-6
 - identifying, 2-10
 - identifying on UNIX, 2-10
 - CTL1 database object, 2-10
 - CTL2 database object, 2-10
 - DR database object, 2-10
 - INDX1 database object, 2-10
 - LOG*thread_id_number* database object, 2-10
 - OEMREP1 database object, 2-10
 - RBS1 database object, 2-10
 - SYS1 database object, 2-10
 - TMP1 database object, 2-10
 - USR1 database object, 2-10
 - identifying on Windows NT, 2-11
 - db_name_ctl1* symbolic link, 2-12
 - db_name_ctl2* symbolic link, 2-12
 - db_name_dr* symbolic link, 2-11
 - db_name_indx1* symbolic link, 2-12
 - db_name_logthread_id_number* symbolic link, 2-12
 - db_name_oemrep1* symbolic link, 2-12
 - db_name_rbs1* symbolic link, 2-11
 - db_name_sys1* symbolic link, 2-11
 - db_name_tmp1* symbolic link, 2-11
 - db_name_usr1* symbolic link, 2-12
- INDX1 tablespace, 2-6
- minimum size for Oracle Database Configuration Assistant, 2-6
- number required, 2-6
- OEM_REPOSITORY tablespace, 2-6
- RBS tablespace, 2-6
- redo log files, 2-7
- setting DBCA_RAW_CONFIG environment variable, 2-8, 2-13, 4-3
- setting up, 2-6 to 2-12
- suggested file names on UNIX, 2-7

- SYSTEM tablespace, 2-6
- TEMP tablespace, 2-6
- USERS tablespace, 2-6
 - verification, 3-5, 3-6
- RBS tablespace raw device, 2-6
- RECOVERY_PARALLELISM parameter, 6-10
- redo log files raw devices, 2-7
- redo log files, described, 1-12
- REG_MULTI_SZ, C-5
- registry
 - DefinedNodes value, 4-6
 - PM\DB_NAME key, C-2, C-5
- REMOTE, 6-4
- REMOTE_LOGIN_PASSWORDFILE
 - parameter, 6-4
- repository database
 - defined, 2-4, 5-3
 - installing, 5-10
- Repository Login dialog box, 5-31, 5-36
- resources, 1-15
- .RHOSTS file, 2-14
- rollback segments, 1-12
- ROLLBACK_SEGMENTS parameter, 1-12, 6-4, 6-10, 6-11
- rolling upgrades, 3-8
- ROW_LOCKING parameter, 6-11

S

- security login, 5-16
- SERVICE, 4-29
- service
 - OracleHOME_NAMEAgent, 5-13
 - OracleHOME_NAMEDataGatherer, 5-30
 - OracleHOME_NAMEManagementServer, 5-14
 - OracleServiceSID, 4-9
- service discovery
 - See auto-discovery
- SERVICE_NAME parameter in TNSNAMES.ORA file, 4-29
- SERVICE_NAMES parameter in INITDB_NAME.ORA file, 6-7
- SERVICES.ORA file, C-9
- SETLINKS utility, 2-9
 - DB_NAME.TBL file, 2-9
 - OP.TBL, 4-3
 - TEMPLATE.TBL, 2-13, 4-3
- SGA, defined, 1-9
- shared disk storage subsystem, 2-5
- shared Oracle home, B-12
- shutdown of the Oracle Parallel Server
 - database, 6-24
 - setting Oracle Parallel Server parameters, 6-39
 - with SYSDBA privilege, 5-28
 - with SYSOPER privilege, 5-28
- shutdown results, 6-24
- shutdown types, 6-22
 - abort, 6-22
 - immediate, 6-22
 - normal, 6-22
 - shutdown database and other services, 6-22
 - shutdown database only, 6-22
- SID, 4-8, C-6
 - defined, 1-11
 - value, 3-4, 6-3
- SID Prefix, 3-4, 4-8
- SID_DESC parameter, C-6
- SID_LIST_listener_name parameter, 4-12, 4-37, C-6
- SID_NAME parameter, C-6
- SID_startup, 4-29, C-8
- SIDALRT.LOG file, C-12
- SIDBSP0.TRC file, C-12
- SIDDBWR.TRC file, C-11
- SIDLCKN.TRC file, C-12, C-13
- SIDLMDN.TRC file, C-12
- SIDLMON.TRC file, C-12
- SIDP00N.TRC file, C-12
- SIDSMON.TRC file, C-11
- SMON process, 1-10
- software requirements, 2-2, 2-3
- SQLNET.ORA file, 4-13
 - configuring, 4-32
 - NAMES_DIRECTORY_PATH parameter, 4-32

- starting
 - Console, 5-15, 6-14
 - listener, 4-9, 4-18, 4-39
 - Management Server, 5-14, 6-14
 - Oracle Data Gatherer, 5-30
 - Oracle Enterprise Manager, 5-13
 - Oracle Intelligent Agents, 5-13, 6-14, B-12
 - Oracle Parallel Server database, 4-9, 6-18, 8-5
 - Oracle Performance Manager, 7-4
 - from Console, 5-32
 - standalone, 5-31
 - the Console, 6-14
- startup
 - results, 6-24
 - setting Parallel Server parameters for, 6-37
 - SYSDBA privilege, 5-28
 - SYSOPER privilege, 5-28
- Startup (START)
 - defined, 1-4
 - described, 1-8
 - on Windows NT, 1-4
- startup types, 6-19
 - force, 6-19
 - mount, 6-19
 - no mount, 6-19
 - open, 6-19
 - restrict, 6-19
- Status Details tab, 6-25
 - from Edit Parallel Server dialog box, 6-29
 - from Parallel Server Operation Results dialog box, 6-25
- Status tab
 - from Edit Parallel Server dialog box, 6-28
- stopping
 - listener, 4-16, 4-37
 - Oracle Parallel Server database, 6-21
- SYSDBA privilege, 5-21, 5-28, 6-38
- SYSDBA privilege, granting, 5-26
- SYSOPER privilege, 5-21, 5-28, 6-38
- SYSOPER privilege, granting, 5-26
- System Global Area (SGA), defined, 1-9
- SYSTEM tablespace raw device, 2-6

T

- Tasks tab
 - from Create Job property sheet, 6-35
- TEMP tablespace raw device, 2-6
- TEMPLATE.TBL file, 2-13, 4-3
- thread ID, 1-12
- THREAD parameter, 1-12, 6-4, 6-10
- TNS_ADMIN environment variable, B-5, B-6
- tns_admin parameter, B-5, B-9
- TNSNAMES.ORA file, 4-13, 5-7, 5-11, 5-29
 - BACKUP parameter, 4-44
 - configuring, 4-20
 - configuring transparent application failover, 4-43
 - FAILOVER parameter, 4-35, 4-41
 - FAILOVER_MODE parameter, 4-35, 4-43
 - how used for auto-discovery, C-7
 - INSTANCE_NAME parameter, 4-29
 - LOAD_BALANCE parameter, 4-25, 4-41
 - METHOD parameter, 4-45
 - SERVICE_NAME parameter, 4-29
 - TYPE parameter, 4-44
- trace files, C-11
 - background thread trace files, C-11
 - error call trace stack, C-13
 - ORAXXXXX.TRC, C-12
 - SIDALRT.LOG, C-12
 - SIDBSP0.TRC, C-12
 - SIDDBWR.TRC file, C-11
 - SIDLCKN.TRC, C-12, C-13
 - SIDLMDN.TRC, C-12
 - SIDLMON.TRC file, C-12
 - SIDP00N.TRC, C-12
 - SIDSMON.TRC file, C-11
- transparent application failover, 4-35
 - configuring, 4-43
 - with connect-time failover, 4-43
- troubleshooting, C-1 to C-14
 - auto-discovery, C-3
 - Oracle Database Configuration Assistant, 4-4, 4-6
 - trace files, C-11
 - TYPE parameter, 4-44

Typical installation, 3-5
Typical installation type, 3-2, 4-2

U

upgrading, Oracle Parallel Server issues, 3-8
uppercase, defined, xvi
User Manager, for Oracle Enterprise Manager, 5-22
user trace files, C-12
USER_DUMP_DEST parameter, C-12
USERS tablespace raw device, 2-6

V

V\$ACTIVE_INSTANCES, 4-11
V\$ACTIVE_INSTANCES table, 6-28
V\$PQ_SYSSTAT table, 1-22
VD-4520 error, 5-19
vppcntl -start command, 5-30
vtm command, 5-31