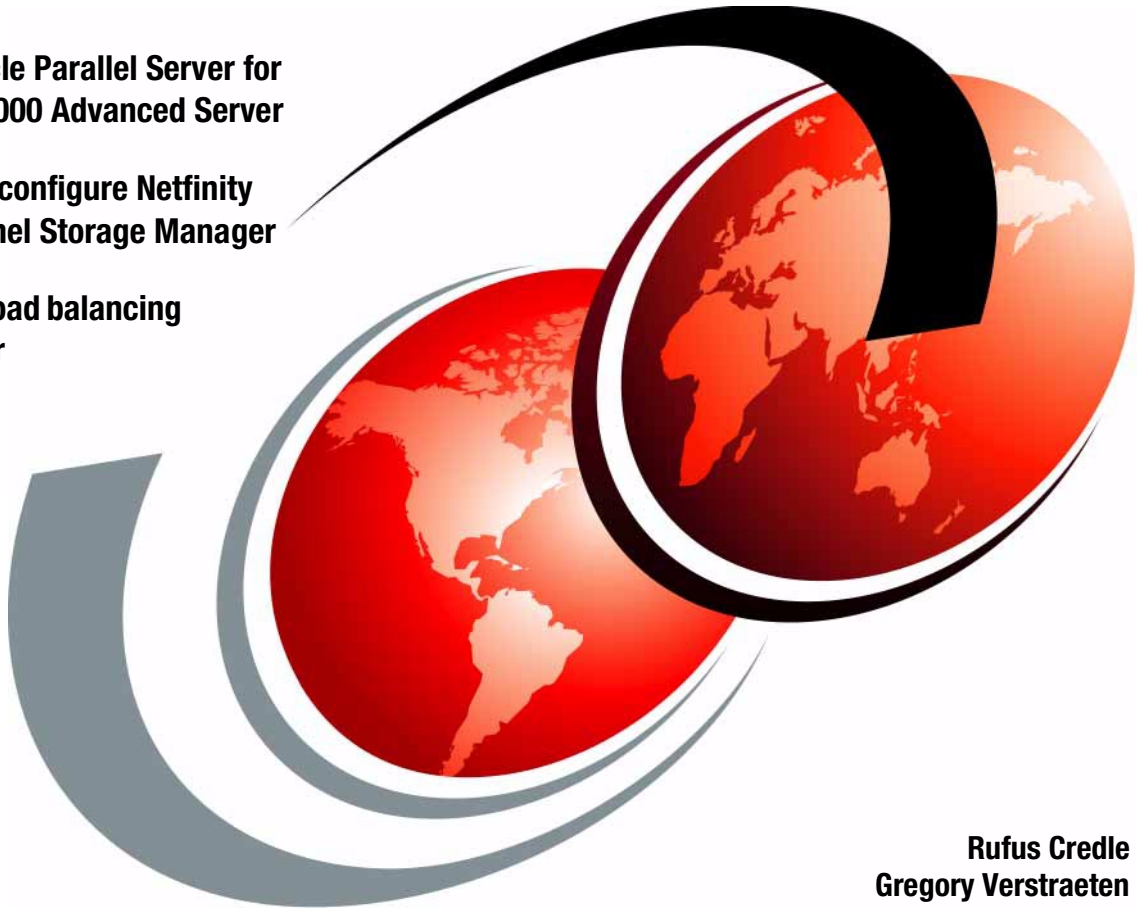
Oracle Parallel Server and Windows 2000 Advanced Server on IBM Netfinity

- Set up Oracle Parallel Server for Windows 2000 Advanced Server
- Set up and configure Netfinity Fibre Channel Storage Manager
- Configure load balancing and failover



Rufus Credle
Gregory Verstraeten

Redbooks

ibm.com/redbooks

SG24-5449-01

International Technical Support Organization

Oracle Parallel Server and Windows 2000 Advanced Server on IBM Netfinity

October 2000

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix B, “Special notices” on page 175.

Second Edition (October 2000)

This edition applies to IBM xSeries and Netfinity systems preparing for the installation of Oracle8i Parallel Server, Oracle8i Enterprise Edition, and Microsoft Windows 2000 Advanced Server.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HQ7 Building 678
P.O. Box 12195
Research Triangle Park, NC 27709-2195

Contents

Preface

This redbook provides an overview of the recent version of Oracle8*i* Parallel Server (OPS) for Windows 2000 running on IBM Netfinity systems. In addition, we introduce the new brand of Intel-based servers known as xSeries.

First you will be given an introduction to Oracle Parallel Server solution where we explain the solution provided by IBM. Next, we provide an understanding of the Oracle and Oracle Parallel Server architecture. Then we discuss the planning phase, the hardware and software requirements, and most important, how to configure your hardware and software.

Once the hardware components and the operating system are installed, we discuss OPS software installation in detail, including the installation of the Operating System Dependent (OSD) layer, and the installation of Oracle8*i* and Oracle Enterprise Manager. In order for you to test and verify the configuration, we also create a database and test it in parallel mode.

This redbook includes the configuration of OPS in a Netfinity Fibre Channel High Availability environment, the installation and configuration of Netfinity Fibre Channel Storage Manager, and the setup and configuration of load balancing and failover.

It is assumed that the reader of this redbook has some experience in using IBM Netfinity Fibre Channel, Microsoft Windows 2000 Advanced Server, Oracle8*i* and Oracle Parallel Server. In addition, a Database Administrator should be available for assistance.

Presently, IBM has certified a two-node IBM Netfinity 8500 configuration with Oracle Parallel Server Version 8.1.6 with patches 8.1.6.1.1 and 8.1.6.1.2 running Windows 4.0 Enterprise Edition with SP5. However, the IBM OPS certification for Windows 2000 had not been completed at the time this redbook was written.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Rufus Credle is a Senior I/T Specialist and certified Professional Server Specialist at the International Technical Support Organization, Raleigh Center. He conducts residencies and develops redbooks about network

operating systems, ERP solutions, voice technology, high availability and clustering solutions, IBM and OEM business applications, all running on IBM xSeries and Netfinity servers. Rufus's various positions during his IBM career have included assignments in administration and asset management, systems engineering, marketing and services. He holds a BS degree in Business Management from Saint Augustine's College. Rufus has been employed at IBM for 20 years.

Gregory Verstraeten is an Oracle product specialist at the Netfinity Technical Support Center in EMEA PSSC in Montpelier, France. He works within the EMEA Oracle/IBM Joint Solutions Center and the EMEA ATS team. Gregory works on designing and sizing complex architectures, supporting benchmarks and presenting workshops involving Oracle products on the Netfinity platform.

Thanks to the original authors of this redbook subject, which was titled *Implementing Oracle Parallel Server on Netfinity Servers,* SG24-5449-00:

Jakob Carstensen, Benoit Iche, and Luke Shutler

Thanks to the following people for their invaluable contributions to this project:

Christopher McCann, PSG World Wide Support Center
Alexander Pope, PSG World Wide Support Center
Roger Bullard, PSG World Wide Support Center
Dennis Hunter, PSG World Wide Support Center
Michael Plean, Netfinity Oracle/BAAN Technical Alliance Manager
Bruce MacKenzie, Cluster System Management
Kristin Parrish, Clustering Platforms
Parmjit Singh, Database Performance Analysis
Steve Britner, PCI Clustering Course Development
Don Flinchum, Netfinity Information Development
Jerry Kaplan, Cluster Enablement
Bob Peckham, NT Cluster Server Analyst
Greg Moore, Netfinity Cluster Project Manager
Bo Brun, PCI Instructor
Stuart Waltz, Netfinity Technology Center
Paul Chenger, Netfinity Technology Center
Don Roy, IBM Netfinity Marketing
Bruno Lahousse, EMEA PSSC, Montpelier, France
EMEA Oracle/IBM Joint Solutions Center

Thanks to the following people from the ITSO Center in Raleigh:

Gail Christensen, Editor
Margaret Ticknor and Michael Haley, ITSO Support

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in “IBM Redbooks review” on page 187 to the fax number shown on the form.
- Use the online evaluation form found at **ibm.com**/redbooks
- Send your comments in an Internet note to redbook@us.ibm.com

Chapter 1. Introduction

In April 1999, IBM announced its solution for an eight-node cluster using Windows NT and Oracle8*i* Enterprise Edition and Oracle Parallel Server. This configuration (the first of its kind in the industry) broke new ground in scalability and high availability. Today, with the release of Oracle8*i* Enterprise Edition 8.1.6 and Oracle Parallel Server support for Windows 2000 Advanced Server, a new solution has arrived that is designed to run on the IBM Netfinity 8500R.

This redbook takes the reader through the implementation and configuration of Oracle8*i* 8.1.6 Enterprise Edition, Oracle Parallel Server and Windows 2000 Advanced Server running on the IBM Netfinity 8500R using the latest IBM Fibre Channel technology.

1.1 Introducing xSeries family of servers

IBM @server xSeries is the new IBM Intel server brand. IBM xSeries are Intel processor-based servers with X-architecture technology enhancements, for a level of reliability, performance and manageability previously out of reach for other industry-standard servers. This represents a full circle of technological evolution for the Netfinity heritage in X-architecture, which is based on technologies derived from the IBM ES, RS, and AS series servers, bringing mainframe-category technology to the industry standard architecture. Also, NUMA-Q will be aligned with xSeries to ensure IBM resources are focused most effectively on the Intel marketplace.

IBM xSeries are available in the following four categories:

- Point Solution Servers
- Universal Servers
- Rack Optimized Servers
- Extremely Scalable Servers

For more information on the IBM xSeries servers, visit the Web site at:

`http://www.pc.ibm.com/us/eserver/xseries/`

1.2 IBM xSeries X-architecture and Oracle: A powerful partnership

In today's business environment the competitive edge is increasingly dependent on a sound IT strategy, a strategy capable of meeting business-critical demands.

The challenge for IT companies is to supply businesses with cutting-edge technologies and powerful, robust computing power at an affordable price. Today's businesses are running mission-critical databases and need to keep these databases up and running 24 hours a day, 7 days a week, 365 days a year.

Such challenges require enterprise computing solutions that provide high availability, scalability, and manageability. These attributes, traditionally delivered through mainframe and midrange systems, are now met through the partnership of IBM's xSeries and Netfinity X-architecture and Oracle Parallel Server software.

IBM X-architecture is a design blueprint that leverages existing, innovative IBM technologies to build the most powerful, scalable and reliable Intel processor-based servers for your business whether you support ten or tens of thousands of users.

As the world's leading server manufacturer, IBM's goal is to provide the greatest capability possible in an industry-standard server at a price point appropriate for your business.

No other company has:

- The availability characteristics achieved by zSeries (proven 99.999% availability)
- The scalability of the pSeries (512 - 1024 node clusters of 8-way and soon, 12-way SMP systems)
- The solution relationships (25,000 application solutions) and self maintaining capabilities (for example, autotuning and autoconfiguration) of the iSeries
- The breadth of storage capabilities from SCSI to Fibre Channel to SSA to network storage, not to mention the skills and experience to integrate these technologies

Because Netfinity engineers are able to capitalize on the vast storehouse of proven technologies developed for the zSeries, pSeries, iSeries platforms, IBM can deliver the benefits of its leadership to xSeries and Netfinity servers

and to your business. IBM also works closely with Intel, Microsoft, Novell and SCO, among others, so that any capability introduced on the xSeries and Netfinity server platform is tested for compatibility with your technology investments, allowing you to build a technological advantage for your business.

1.3 High availability

One of the primary motivators for such a partnership is to meet business requirements for a solution that could approach continuous availability within the Windows 2000 and Intel environment. The three steps in achieving this are:

1. Reduce the probability of failure

   The chances of failure can be reduced by improving the reliability of hardware components and software. This is an on-going process and accomplished through new technology, error-correcting code and other methods.

2. Minimize the effects of failure

   This can be tackled in a number of ways, usually involving the duplication of “at risk” components within the server, such as disk drives, network adapters, power supplies and cooling fans. With Netfinity OPS solution this redundancy is taken further with the duplication of Fibre Channel adapters, RAID controllers and hubs. In addition to this, Netfinity OPS solution is designed to tolerate complete node failures.

3. Perform scheduled maintenance and upgrades without down time

   Netfinity OPS solution enables the node to be powered off while repairs or upgrades to hardware and software are being carried out on the system. This means that planned and unplanned maintenance can be carried out without failure of the cluster and without significant impact to the attached users.

1.4 High availability on Oracle databases

The Oracle offering for clustering is composed of two products: Oracle Fail Safe (OFS) and Oracle Parallel Server. OFS offers high availability on Microsoft Windows NT and 2000 servers. OPS offers high availability *and* scalability on Microsoft Windows NT and 2000 servers.

Oracle Fail Safe is layered over Microsoft Cluster Services and is tightly integrated with the "shared nothing" cluster environment. In a shared nothing

cluster, resources (such as disks) are owned and accessed by only one node at a time. If one server fails, the other server of the cluster can handle the user workload. But there is no performance scalability; the maximum computing power is a single system computing power.

Oracle Parallel Server implements a "shared data" model, which by contrast allows multiple servers to simultaneously access the disk storage. Each server holds a database instance. This allows a single workload to scale across multiple systems.

The OPS technology brings many advantages:

- Availability is increased because each server can function independently and provide database access in case of the failure of any or all of the other servers.
- More users can be supported because each server can support the maximum number of connections allowed by a single server.
- Large amounts of data can be managed efficiently.
- Queries and data manipulation can spread across multiple servers to reduce their overall processing time.

1.5 Oracle Parallel Server and IBM Netfinity

Oracle Parallel Server has been deployed for nearly a decade at many customer sites around the world on a variety of platforms. Today, Oracle Parallel Server is a mature and reliable product. In 1997, Oracle launched OPS V8.0.4 for Windows NT clusters. That was the first version of OPS available for Windows NT Server. IBM announced in June 1998 a six-node configuration on OPS software. This was the first time that IBM introduced a Clustered Fibre Channel solution. The Clustering Middleware called Phoenix on the Windows NT platform and IBM Netfinity supported up to six nodes in a solution.

Presently, IBM supports Oracle Parallel Server on two 8500Rs running Windows NT 4.0. This solution is known as IBM Netfinity Advanced Cluster Enabler for Oracle Parallel Server V2.1 using Oracle Parallel Server 8.04, 8.05 or 8*i* 8.1.6 and Fibre Channel Storage. IBM offers this solution to enhance the availability and scalability of Oracle database applications. All the hardware components included in this configuration are IBM components and the OSD component is called the IBM Netfinity Advanced Cluster Enabler 2.1.

The configuration for the IBM OPS solution mentioned in this redbook includes the following configuration: IBM Netfinity 8500R clustered with Fibre Channel architecture running on Oracle Parallel Server 8*i* R2 (8.1.6). In this redbook, we have configured and run failover tests using Oracle Parallel Server 8*i* R2 (8.1.6) and Windows 2000 Advanced Server running on a two-node, IBM Netfinity 8500R (eight processors each) cluster with Fibre Channel architecture and an EXP200 External Storage Expansion Unit.

The Oracle Parallel Server, using its leading Database Management System (DBMS) technology, complements IBM Netfinity's high availability designs and in doing so provides increased levels of availability for mission-critical applications. This is achieved through the clustering of multiple nodes, with each node running a separate instance of the database. In this scenario, two or more nodes can access a single database. This provides benefits both in scalabilty and availability in comparison to a simple failover cluster:

- Node failures only affect the subset of users attached to the failed node. The failure does not affect users connected to other nodes in the cluster.
- Users attached to the failed node fail over to the surviving nodes in the cluster. Because Oracle Parallel Server is already running on the other nodes this failover is fast and reduces disruption to connected users.
- All nodes within the cluster are active. Because of this, system resources are better utilized and load balancing across nodes takes place, giving greater efficiency than an active - passive cluster environment.
- The solution is scalable. However, IBM currently supports a two-node configuration using Windows NT 4.0.

Another major area of fault tolerance and high availability within the Oracle8*i* is the use of Transparent Application Failover (TAF). TAF provides an application with the ability to automatically reconnect when an instance is lost due to hardware or software failure. Not only is the attached user reconnected to a new instance, but any queries being generated at the time of the crash are also restarted and the application resumes from the point of failure.

Chapter 2. Overview of OPS architecture

This chapter includes how applications interact with OPS, the hardware topology for OPS and an overview of the Oracle database and OPS architecture.

2.1 Applications using Oracle Parallel Server

Oracle Parallel Server is a resource-sharing system that increases availability and performance by partitioning the workload across multiple servers of a cluster (nodes).

OPS effectiveness is provided by:

- Synchronization of tasks and subtasks to prevent steps with serialized dependencies from being completed in the wrong order
- Locking to avoid conflicted use of the same resource by different tasks
- Messaging to enable synchronization and locking

Availability will always be increased with OPS. In case of a node failure, the workload of this node will be handled by the other node of the cluster, that is the failover.

To increase performance, the workload must be partitioned. A single application must have subtasks that can be worked on simultaneously and independently, or multiple tasks must be able to complete their work without requiring the resources used by other tasks. Applications must be designed to be scaleable. A bottleneck will occur in the system in which every node is updating the same data most of the time.

Performance is made of two elements, scaleup and speedup. Scaleup is the capability to handle more workload when the number of servers increases. Speedup is the capability to execute the workload faster.

With the use of OPS, certain applications will experience increased performance more than others.

- Data warehousing applications

  OPS can increase both speedup and scaleup on these applications, because these applications essentially support large tasks that do not require exclusive access (read queries), and because long-running queries can be run concurrently.
- OLTP (on-line transaction processing) and Internet applications

These applications are characterized by short transactions that cannot be passed down further, and therefore no speedup can be achieved. However, by deploying greater amounts of resources, a larger volume of transactions can be supported without compromising the response if the application affects different sets of data (partitioned database). If OLTP applications are not well designed, there may be performance degradation because of the increased overhead required for synchronization.

OPS will cease to be advantageous when the cost of synchronization becomes too high, and the throughput has decreased. If many end users connected to a large number of nodes are modifying a small set of data, then synchronization is likely to be very high. However, if end users are just reading the data, less synchronization is required. Though note that this is true only if the data is absolutely static.

Most OLTP applications are not update intensive. For example, travel reservation systems spend most of their time on flight and fare searches, not bookings. When changes to existing records are made, more time can be taken to perform index scans searching for the desired records than to make the changes themselves. Yet, when updates are required, it is important to prevent the same blocks from being modified by different instances at the same time, or almost the same time. This is done by partitioning the data, users, applications, or functions in such a way that each instance works with a subset of the data that is not required by any of the other instances. If this is done, any version of Oracle Parallel Server will scale as more instances are added.

Partitioning can be done in several ways:

- Data partitioning: the tables and indexes are partitioned (for example, per periods), and instances access different partitions.
- Users partitioning: users from different departments of a company are accessing the same application. Users from each department are connected to a department-specific instance. This will reduce the need of synchronization between instances.
- Applications partitioning: financial and manufacturing modules from an ERP would be connected to dedicated instances, because these modules share few data.
- Functions partitioning: in big applications such as the financial module of an ERP, smaller parts of the application associated to functions (for example, purchasing, payroll, account receivable...) can be connected to dedicated instances.

If partitioning is not possible or is not adequately implemented, the ability to scale to multiple instances is limited. In the very early releases of Oracle Parallel Server, non-partitioned applications would not scale if they exceeded a very low rate of data modifications.

With the introduction of Oracle8, a number of performance enhancements in the inter-instance locking mechanisms enabled applications with light-to-medium transaction rates to scale, even without partitioning. Oracle8*i* introduced even more sophisticated technology that enables applications with even higher transaction rates to scale without partitioning. This technology is being extended to provide scalable solutions regardless of whether any partitioning is possible or enabled.

2.2 OPS hardware topology

In this section, we discuss the current supported OPS topology, IBM Netfinity Advanced Cluster Enabler for Oracle Parallel Server V2.1 and our setup of OPS and Windows 2000 Advanced Server.

To implement an OPS solution you can have two or more nodes. However, in this section, we describe in our example the use of two nodes. One node is connected to the other by high-speed dedicated Ethernet adapters. This connection is used for communication between the Netfinity 8500Rs and the different Oracle instance.

In our configuration, two Netfinity Fibre Channel Adapters are installed per Netfinity 8500Rs. Each adapter is connected to a different Netfinity Fibre Channel hub which is connected to a Netfinity Fibre Channel RAID Controller, which is housed in a Netfinity Fibre Channel Controller Unit. This configuration is a redundant Netfinity Fibre Channel setup. If a Fibre Channel adapter, a hub or a Netfinity Fibre Channel RAID Controller goes down, you can still reach your data.

You can increase the availability of your configuration by distancing the servers and the storage expansion units. Connections from the host adapters and the RAID controller units are all short-wave connections. This means the maximum distance between both is 500 meters. By adding long-wave Gigabit Interface Converters (GBICs) in the hubs, which support cable lengths of up to 10 km, you can obtain a fully redundant cluster.

In our configuration, we will set up and configure a topology similar to Figure 1 on page 10. Yet, instead of running OPS on Windows NT 4.0, we will run OPS on Windows 2000 Advanced Server with IBM Netfinity Fibre Channel. For Windows 2000 Advanced Server, it is a requirement that we use Netfinity

Fibre Channel Storage Manager 7.01. In addition, all Fibre Channel adapters and controller units should use the latest firmware and drivers required for Windows 2000.

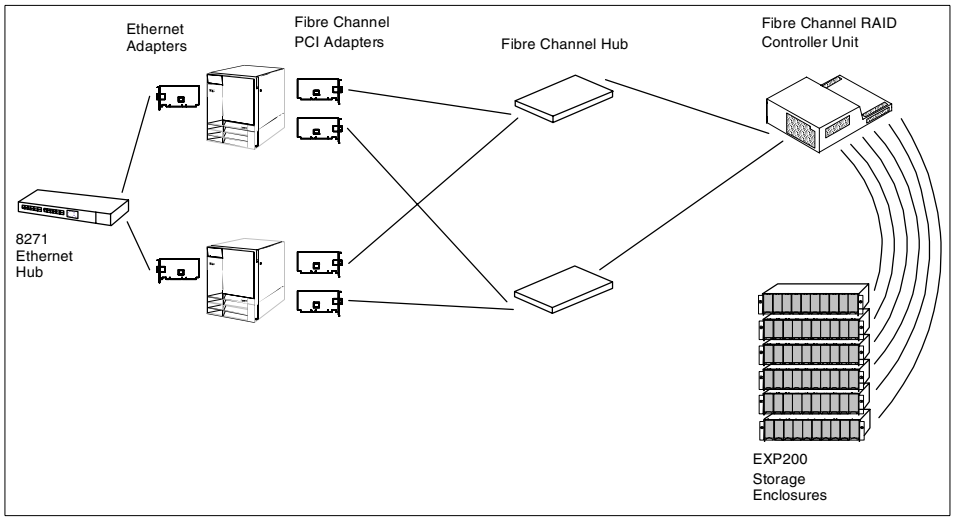


Figure 1. OPS two-node configuration

Figure 1 shows an OPS configuration that includes two Fibre Channel adapters in each Netfinity 8500R server, each connected to a different Fibre Channel hub. You can connect up to six Netfinity EXP200s or EXP300s to one Fibre Channel controller unit. If you need more than 60 hard disk drives, it's possible to add another Fibre Channel controller unit. If you want to add another Fibre Channel controller unit attached to the six nodes configuration, you must also add two Fibre Channel hubs because there are only seven ports per Fibre Channel hub.

The present IBM-certified configuration consists of the following:

- Two Netfinity 8500R servers
- A Netfinity Fibre Channel Controller Unit with two RAID controllers
- Two Vixel 7-port hubs and loops
- Netfinity Fibre Channel PCI Adapter
- Optical cabling, GBICS
- Netfinity EXP200
- Windows NT 4.0 E.E.

2.2.1 Cluster

To install the IBM Netfinity Advanced Cluster Enabler for Oracle Parallel Server V2.1, Windows 2000 Advanced Server and IBM Netfinity Fibre Channel Storage Manager 7.01 must be installed. This software allows you to configure and administer your Fibre Channel disk. On top of these items, you install Oracle8*i* Parallel Server.

The IBM Netfinity Advanced Cluster Enabler for OPS V2.1 is the Operating System Dependent (OSD) layer. OSD is the interface between Oracle and platform-specific cluster services to provide the architected functions necessary for the operation of Oracle Parallel Server. The OSD layer consists of several software components distinct from Oracle Parallel Server. These components provide key services required for proper operation of the Oracle Parallel Server options. In total, the OSD components provide what can be viewed as the Oracle Parallel Server interface to the cluster and its distributed services. The OSD layer's most important components are the Cluster Manager (CM), the Interprocess Communication (IPC) and the IO (input/output).

To order the IBM Netfinity Advanced Cluster Enabler for OPS V2.1, contact your IBM Sales Specialist or Business Partner.

The OSD is developed by Oracle, while IBM has wrapped its installation code around it. You can find it on the Web at:
`http://www.pc.ibm.com/us/netfinity/parallel_server.html`.

2.2.2 Interconnect

In an OPS solution, all the nodes are connected to an external disk enclosure via a Fibre Channel interface and together via Ethernet adapters. These two connections are needed to allow multiple instances to access a shared database. OPS allows separate Oracle instances running on different nodes and all instances can share the same database (execute transaction concurrently) and each instance can have multiple concurrent users.

The interconnection between the different nodes is needed for the coordination of each node accessing the shared database, which allows data consistency and integrity.

2.2.2.1 Hardware interconnection

For the communication between servers, we need an interconnect of dedicated 10/100 Ethernet adapters utilizing a crossover cable. This connection should be as fast as possible. Therefore, set adapters to operate at the 100 Mbps rate.

The availability and the speed of this connection is very important for the cluster. It allows communication between the different instances.

For many cluster configurations, existing industry-standard technologies such as Ethernet and ATM switching provide ample bandwidth for inter-node communications and messaging. However, as clusters scale to 8, 16, 32 or more nodes per cluster, optimized interconnect technologies called System Area Networks (SANs) will become increasingly more important. SANs provide inter-node communication with extremely high bandwidth and very low latency (delay between the directing signal and the availability of the required signals from memory).

2.3 Oracle concepts

In this section, we describe the relational database concepts, then we explain the following Oracle files:

- Data files
- Redo log files
- Control file
- Initialization parameter file (init.ora)

Also, we give an outline of the architecture of an Oracle instance.

2.3.1 Relational database concepts

In the following we provide an overview of the features that are common to all Oracle databases. The following topics are discussed:

- Relational databases
- Database structure
- Structured Query Language
- Client/server architecture

2.3.1.1 Relational databases

In this type of database, the information is stored in tables. Each table has a name and is logically represented as being made up of rows and columns. Tables can be related to each other if they contain columns with common types of information. See Figure 2 on page 13.

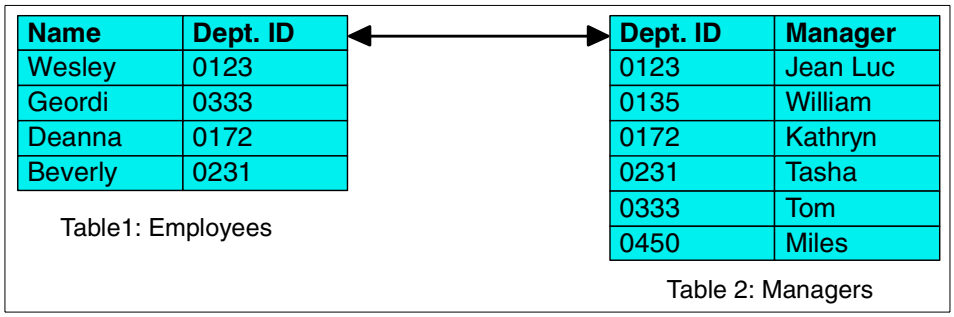


Figure 2. Relationship between two tables

Although the information stored in each table may be independent, there is obviously a link between the two tables shown. The department ID in the Employees table is “related” to the department ID in the Managers table. To find the name of the manager of a particular employee, look up his or her department ID (in the Employees table) and then find the manager of this department (in the Managers table).

Relational databases are data driven; that is, the tables and relationships between them are defined once, and then only the data changes over time. Data changes have no impact on the applications using them.

2.3.1.2 Database structure

The structure of a relational database can be divided into three parts:

- Physical
- Logical
- External

The physical part is made up of the files, the directories, and the physical storage elements.The logical part is made up of the objects that are referenced by the database. These include tables, tablespaces, and other elements that make up the relational database model. The boundaries of logical structures are independent of physical space allocation. The external part represents the data as seen by the users. This includes views of the data, the clustering of the information, and accessibility to the various tables.

This three-tiered approach allows for an independence between the data and the means of storage of the data, and it allows for different types of access to the data depending upon the needs of the user.

2.3.1.3 Structured Query Language (SQL)

SQL is a non-procedural language that is used for most database actions. It was developed by IBM Research and has since been refined by ANSI.

SQL Statements

The SQL language consists of about 30 statements. Each SQL statement begins with a verb (which describes what the statement does) followed by one or more clauses.

The main types of SQL statements are:

- Queries
- Data Manipulation
- Data Definition
- Access Control
- Transaction Control

Queries are statements that retrieve (but do not change) data and begin with the reserved word `SELECT`.

Data Manipulation statements are used to change data by using instructions such as:

- `INSERT` to add new rows of data
- `DELETE` to delete rows of data
- `UPDATE` to change column values in existing rows

Data Definition statements are used to create, alter and drop database objects. They differ from the previous two types of statements because they require a write access to the data dictionary. Typical reserved words are `CREATE TABLE`, `DROP TABLE` and `ALTER TABLE`.

Access Control statements are used for two types of access:

- Access to the database system
- Access to the database data

Access Control statements such as `GRANT` and `REVOKE` can control user access privileges.

Transaction Control statements are of two types:

- `COMMIT` to end the current transaction
- `ROLLBACK` to abort the current transaction

Context areas and cursors
Each SQL statement is associated with a cursor or context area. This is a memory buffer created on the server that contains the current status of one SQL statement.

A cursor can be thought of as a file handle (or name of the context area) that is opened to gain access to the SQL statement results. The statement is processed or parsed within the cursor, and as long as the cursor remains open, the statement can be re-executed without being re-parsed.

Just as a file handle keeps track of the program's current position within a file, the cursor keeps track of the state or phase of the statement. Cursors are only capable of forward sequential processing. When the statement has been fully parsed and no longer needs to be executed, the cursor is closed either explicitly or by the client terminating the connection to the server.

Statement parsing
An SQL statement is parsed (processed), and a representation of the statements is loaded into the cursor. Parsing consists of:

- Translating the SQL statement
- Loading it into a cursor
- Verifying user access privileges (access control)
- Determining access paths to be used when executing the statement
- Determining resource requirements
- Reserving the resources required

2.3.2 Oracle files

The installation of Oracle consists of the following sets of files: data files, redo log files, control file and init.ora. In an Oracle database, we have both a physical and a logical structure. These files are in the physical structure.

2.3.2.1 Data files

An Oracle database contains one or more data files only. The data file is the same in exclusive or parallel mode. Database files have the following characteristics:

- A data file is associated only with one database.
- The size of a data file cannot be changed after its creation.
- One or more data files can be grouped together to form a tablespace.

Tablespaces
A database is made up of one or more tablespaces. Tablespaces are useful in helping you to organize your data for ease of management, security, and performance. In fact, tablespaces are logical storage units. There are two different types of tablespaces:

- *System tablespace* - is used to store the data dictionary. This holds information such as names of tablespaces and what data files are contained in each one.
- *User tablespace* is used to hold personal data.

Segments
A segment is a generic name given to any object that occupies storage in the database. A segment is made up of a group of extents (contiguous blocks) that are in the same tablespace (but not necessarily in the same data file).

Extents are used to minimize the amount of wasted storage and can grow or shrink as required. Extents themselves are made up of data blocks, which are the smallest pieces that make up an Oracle database and are physically related to the disk partition size. An Oracle database has the following types of segments:

- **Rollback segment**: Whenever data is altered, this change must either be committed or rolled back. The rollback segment holds the previous version of the data being modified; this allows for recovery from aborted or incomplete transactions. Any instance needs at least one rollback segment or it will not be able to start. OPS needs at least as many rollback segments as the maximum number of concurrent instances plus the system rollback segment.
- **Temporary segment**: This is used by Oracle when an SQL statement needs a temporary work area. It is destroyed upon completion of the statement.
- **Data and Index segments**: Data segments store user data within the database. Index segments store indexes used by Oracle to look up data quickly. An index scan of the database is much quicker than a full table scan, where Oracle would look at every row in the database.

2.3.2.2 Redo log files
The redo log files are journal files that record all changes made to the database. These files are in memory (for performance reasons, disk I/O is roughly a thousand times slower than actions in memory), and Oracle can then write the changes to the data files (on disk) at its leisure. Every Oracle database has a set of two or more log files.

These files are used in case a failure (machine, disk etc.) prevents modified data from being permanently written to the data files. During recovery, Oracle will apply the data in these files to bring the database to a consistent state without losing any committed transactions.

Online redo logs
Since these files are open or online during normal operation, they are referred to as the online redo log files. The online redo logs work in a circular fashion. As transactions take place, they are recorded in the first redo log. When this is full, a log switch occurs. Now, transactions are recorded in the second redo log. When this is full, another log switch occurs. All new transactions are once again recorded in the first log, overwriting the previous contents.

Archived redo logs
Oracle offers the possibility of running in either ARCHIVELOG mode or NOARCHIVELOG mode. In ARCHIVELOG mode, the contents of the redo logs are copied to an archive area (on disk) before they are overwritten. These archive files are known as the archived redo log files. Because they are not open during normal operation of the database and are required only during data recovery, they are also known as the offline redo logs.

In NOARCHIVELOG mode, old redo logs are not kept, and the redo logs are simply overwritten.

2.3.2.3 Control file
Every Oracle database has a control file. It is highly recommended that you have more than one copy to guard against data loss. This file records the physical structure of the database and contains the following types of information:

- Name of the database
- Creation date and time
- Location of the database
- Status and state of all the data files
- Location of the redo logs

Whenever there is a change to the structure of the database, the control file is updated.

All OPS instances access the same control files. There is only one control file per cluster. The other control files are just mirrored files.

2.3.2.4 The init.ora file

The init.ora file is a parameter file (text) containing Oracle system parameters. It is provided by Oracle and should be customized for your site. This file is read during database startup to determine the size of the system global area and to locate the control files. The actual name of the init.ora file has the Oracle instance identifier appended to it.

2.3.3 Oracle memory structure

Oracle uses system memory to run user processes, cache data, and indexes to store shared program code. Oracle has two types of memory structures:

- The system global area
- The program global area

2.3.3.1 System global area or shared global area (SGA)

The SGA is an area of shared memory used by Oracle to store data and control information for one Oracle instance. All of the information contained within the SGA is shared by all of the users connected to the instance. The SGA is allocated when the instance starts up and is automatically de-allocated when the instance shuts down. For maximum performance, the SGA should be as large as possible to store a maximum amount of data in memory in order to reduce the disk I/O. The SGA is divided into a different memory structure, which is created during the instance startup and has a fixed size. The SGA is made up of the following key components:

Data buffer cache

The data buffer cache stores the most recently used data blocks, that is those blocks that have been modified but not yet written to disk (dirty blocks) and those that have been written to disk (clean blocks).

Before a user process can access a piece of data, it has to be in the data buffer cache. A least recently used algorithm is used to free up space when new data is requested by a user.

Redo log buffer

Before any transactions can be recorded in the redo logs, they must first reside in the redo log buffer. They are then written to the redo logs by a database background process (log writer). Its size is static.

Shared pool

The shared pool is a cache containing all the parsed SQL statements that are ready to run. This is useful for reducing overhead (memory, processing time,

execution planning time) when multiple applications issue the same SQL statement.

2.3.3.2 Program global area (PGA)

A PGA is allocated when a user connects to the database. The PGA is not shared and contains data and control information for single Oracle server processes or for Oracle background processes. The PGA is made up of the following components:

- Stack space: Memory holding session variables
- Session information: Stored here for dedicated servers and in the SGA for multi-threaded servers
- Private SQL area: Information on binding variables and run-time buffers

2.3.4 Oracle processes

An Oracle instance is composed of an SGA and Oracle processes. It consists of:

- User processes
- Oracle server processes
- Oracle background processes
- Shared memory used by these processes

Figure 3 provides a graphical description of the Oracle architecture:

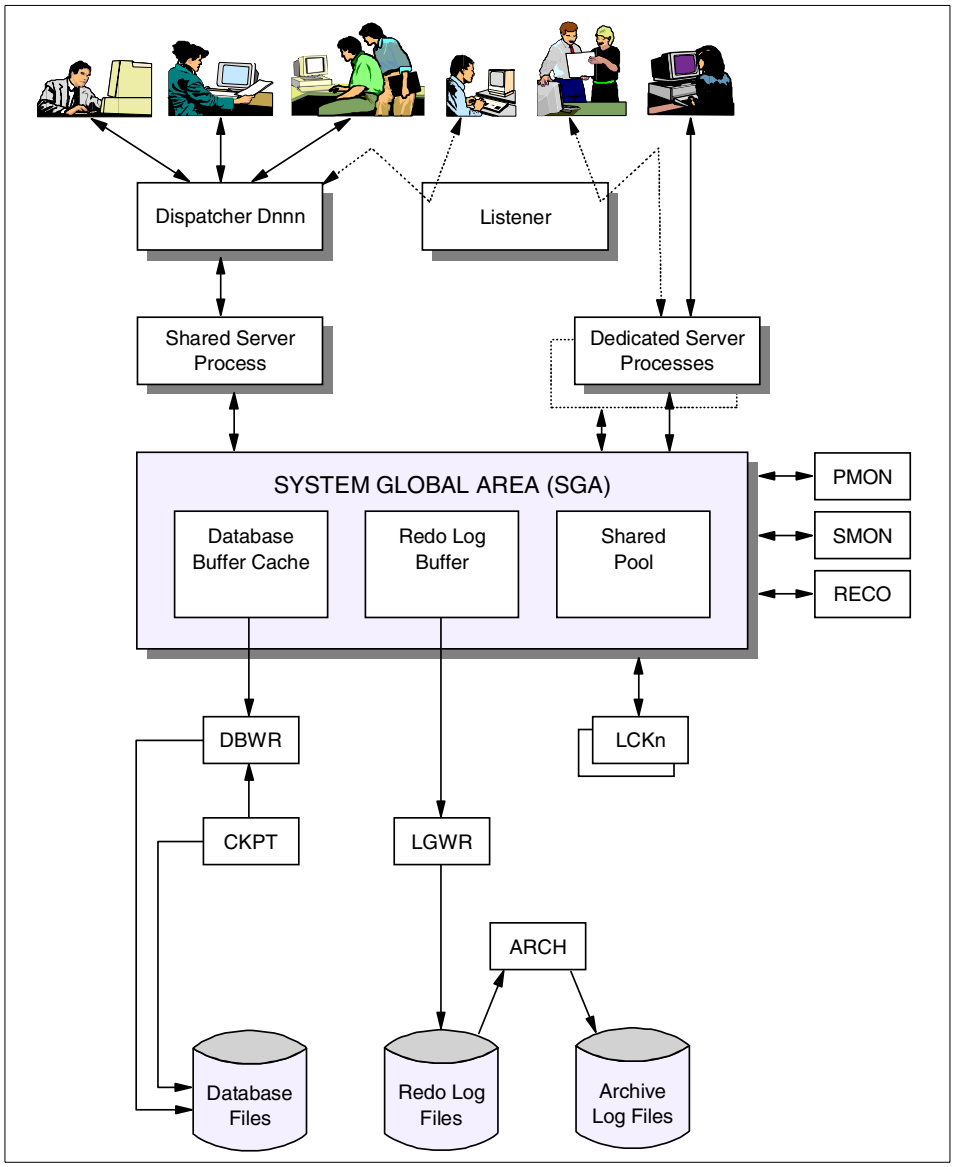


Figure 3. Oracle architecture

2.3.4.1 Main processes

There are four Oracle system background processes that must always be up and running for the database to be usable. They are:

- Database writer (DBWn)

The database writer process is responsible for writing modified (dirty) data blocks from the database buffer cache to the database files. The DBWR does a write only when buffers are needed to read in data. To improve write performance, it's possible to have several DBWn processes (DBW1 through DBW9).

When using the Oracle Parallel Server, the DBWn might be forced to do a write, not because buffers are needed but because another user needs to modify the same buffers. This process is known as pinging and can be responsible for degrading database performance.

- Log writer (LGWR)

  The log writer process writes redo log entries from the redo log buffer to the redo log files on disk.

- System monitor (SMON)

  The system monitor process performs instance recovery at database startup time or, in the case of the Oracle Parallel Server, when another instance belonging to the database has crashed or terminated abnormally.

  SMON also releases temporary segments that are no longer needed, compacts the free space fragments in the database files, and detects deadlock situations.

- Process monitor (PMON)

  The process monitor process keeps track of database processes. If a user process fails, PMON cleans up the cache and frees up any resources that the failed process was using.

  PMON also monitors the dispatcher and shared server processes and restarts them if necessary.

2.3.4.2 Optional system processes

Along with the four processes described above, there are a number of other optional system processes, such as:

- Checkpoint (CKPT)

  A checkpoint is an event in which all modified data blocks are written by the DBWR process to the data files. This usually occurs at a redo log switch. At a checkpoint, all of the database file headers and the redo log file headers are updated to record the fact that a checkpoint has occurred.

  The CKPT is optional. If it is not present, the LGWR process performs the tasks of the CKPT process. It is recommended that the CKPT process be enabled when there are a large number of data files.

- Recover (RECO)

  The recover process is used when there is a failure in a distributed transaction. A distributed transaction is one where two or more locations of the data have to be kept synchronized. In this environment, there may be multiple databases on multiple interconnected servers and either a node or a network fails.

  Any transaction that may have completed in one site but not in another is referred to as in doubt. The RECO process attempts to establish communication with the remote servers. When the connection is reestablished, the RECO process automatically resolves all the in-doubt transactions.

  The RECO process is optional and is needed only in instances that carry out distributed transactions.

- Archiver (ARCn)

  The archiver optional process is used when the data is running in ARCHIVELOG mode and automatic archiving is enabled. ARCH copies the redo entries from the online redo log files to the archive area.

- Parallel server lock processes (LCKn)

  The lock processes are used only if running Oracle Parallel Server. LCKn uses the distributed lock manager for inter-instance locking to prevent simultaneous changes to the same data from different instances.

2.4 Data storage

This section looks at the theory behind the Oracle Parallel Server file hierarchy and how to physically implement raw partitions on the logical partitions under Windows 2000.

The files of a database provide the actual physical storage for database information. All of the databases used allow data to be stored either as files within a file system or directly on the disk as raw data.

In this section we define and discuss the file structure used by Oracle Parallel Server.

2.4.1 File systems

Using file system files may be easier than using raw devices. In this case, the database simply creates a file in the filesystem. However, when using the file system, the database must contend with the disk caching that is inherent in block devices. Block devices do buffered I/O, where data is collected in a

buffer until an entire block can be transferred at one time. Generally, writes need to be done synchronously in order to ensure coherency.

On reads, there are overheads due to the data being read into the disk cache before becoming exploitable by the database. There is also the problem that the files in the file system may not be placed contiguously on the disk, and additional overheads will be incurred due to disk seek times.

2.4.2 Raw partitions

From the Windows 2000 point of view, a raw partition is simply a logical partition (in an extended partition) created with Windows 2000 Disk Manager with no formatting applied.

With Oracle Parallel Server different instances must access shared disks. To do this, we must bypass the Windows 2000 file system by using raw partitions.

A raw partition is an area of contiguous physical and logical disk space that is under the direct control of an application rather than under control of the operating system and file system.

The applications use raw input and output, carrying out a data transfer with every read or write. As data is written directly to the disk, bypassing all operating system disk caching and file system overheads, performance is generally improved. The I/O is between 5 to 10% faster.

For the backup and recovery of a database on raw partitions, Oracle RMAN is the only tool available. File copy will not be possible.

2.5 Oracle Parallel Server concepts

In this section, we describe the OPS concepts.

2.5.1 Parallel databases and parallel processing

These two technologies offer greater advantage for decision support system (DSS) and online transaction processing (OLTP). In this chapter we will see the benefits of these two technologies, how they work and how to implement them.

2.5.1.1 Definitions

The definitions of parallel database and parallel processing are as follows:

Parallel database
A parallel database is designed to run multiple instances that share a single physical database. So when a parallel server is enabled, up to eight instances mount the same database. The workload can be balanced among CPUs and the data can be accessed concurrently.

Parallel processing
The aim of parallel processing is to execute a task more quickly. For that, the large task is divided into smaller tasks executed concurrently on several nodes.

But not all tasks can be divided into smaller ones to reduce the execution. For example, if in a job the tasks are dependent, parallel processing will not improve performance. That means that parallel processing must structure tasks (tasks can be performed concurrently) and preserve task sequencing.

2.5.2 Instances interconnection

The IBM Operating System Dependent (OSD) layer consists of several software components required for proper operation of Oracle Parallel Server. Two important components for the interconnection of the nodes are Cluster Manager (CM) and InterProcess Communication (IPC) (see Figure 4). IPC provides reliable transfer of messages between instances on different nodes and CM discovers and maintains the state of the cluster. IPC is used by the Integrated Distributed Lock Manager (IDLM). The role of IDLM is to coordinate simultaneous access to shared databases in a way that maintains consistency and data integrity. The disk update operations are coordinated by IDLM so that all nodes can see the same data. For example, if an Oracle instance wants to update a cached data block it must enter into a dialog with IDLM to ensure it has the exclusive right to update the block.

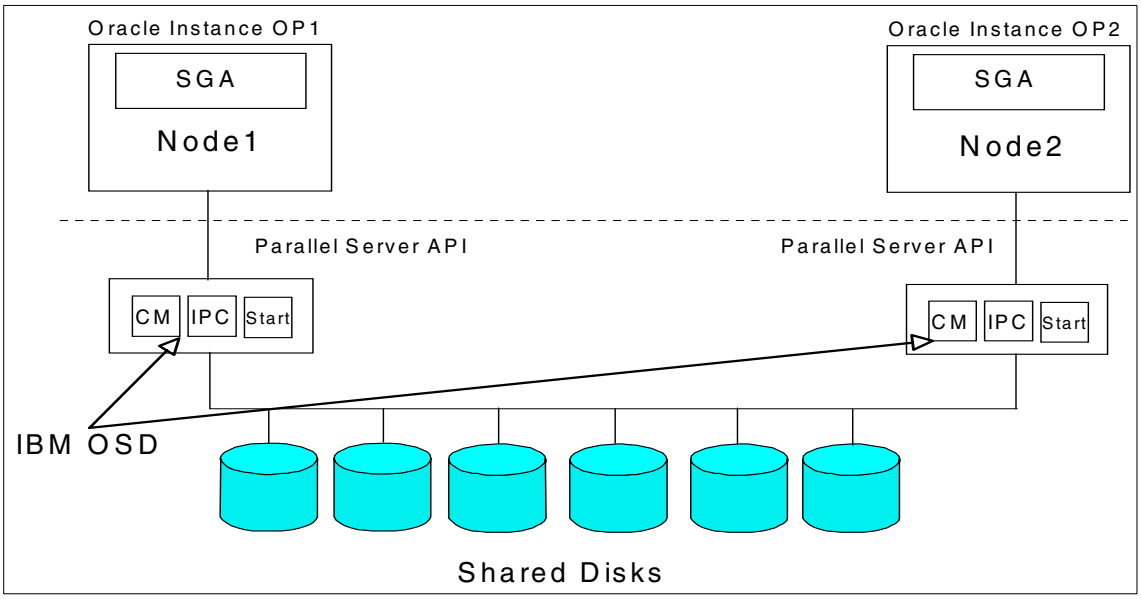


Figure 4. An example of the OSD components in a two-node configuration

2.5.2.1 What is the OSD?

The operating system dependent layer (OSD) developed by Oracle provides key services that are required in order for Oracle Parallel Server to function. The OSD is dependent upon the operating system's services and uses standard networking APIs and Windows 2000's APIs for raw I/O.

For information on the services contained within each Oracle instance please refer to 2.3, "Oracle concepts" on page 12.

To order the IBM Netfinity Advanced Cluster Enabler for OPS V2.1, contact your IBM Sales Specialist or Business Partner.

2.5.2.2 OSD layer components

This section details some of the individual components of the OSD layer and how each component interacts with other services within the OPS framework.

Cluster Manager (CM)

The Cluster Manager (CM) service within the OSD, which replaces the PGMS service that was available in Release 8.0, controls and manages membership characteristics within the cluster. Its primary task is to interface with OPS to control the joining (attach) or leaving (detach) of each node within the cluster. You can say that the CM service is managing cluster membership.

In a failover situation, it is critical that all remaining instances are notified of the failure and that relevant OPS recovery operations can be started.

For example, should any node or instance be considered dead or not functioning correctly, then the CM will signal the new membership to its attached clients (OracleService). Oracle will then decide what to do with the database instance and Oracle processes. When a node fails, clients connected to that node will connect to other instances on the surviving node(s).

This failure is transparent to user applications. CM reconfigures the cluster to isolate the failed node and notifies the Integrated Distributed Lock Manager (IDLM) of the failure.

The CM is itself split into two separate subcomponents:

- node monitor component
- cluster management component

The node monitor component accesses the shared disks and checks the status of the nodes, network, and the CM component itself. The Node Monitor component communicates at a low level among nodes to determine which nodes are alive. The status of the node is communicated back to OPS through the OSD-CM component. At this time, OPS will determine what it needs to do to manage the database disks.

It provides OPS with the modules required for the basic management interface within the cluster and allows OPS to track the membership state of each node. It does this by providing a view of nodes and their member status within the cluster.

The node monitor will monitor the topology of the cluster and polls the nodes for current membership status and reports this information back to the CM component, which will then feed this to OPS.

The Cluster Management component manages instance members. Each instance registers with its database-specific group, which is managed by the Cluster Management component. This information is passed to the Node Monitor and then through the CM to OPS.

Interprocess communication (IPC)

In order for OPS to work within a clustered environment the nodes must be capable of communicating with each other. It is the inter-process communication (IPC) component that regulates and facilitates this communication between nodes and instances.

The IPC defines the protocols and interfaces needed within the OPS cluster environment to transfer messages between instances using an asynchronous queued message model. It uses an efficient communication layer that allows it to transmit and relay messages very quickly. It is this communication layer that the IDLM requires to communicate to all nodes.

2.5.2.3 Input/output (I/O)

The input/output (I/O) component provides Oracle Parallel Server with the interprocess capabilities required to operate in a cluster environment. It is the I/O layer that allows OPS to support simultaneous disk sharing across all nodes.

The I/O service coordinates disk update operations between nodes and ensures that data is kept consistent. This is a complex task that needs to utilize an efficient I/O model. At a high-level view, the shared I/O model used by OPS can be described as distributed disk cache implementation. This is because the key to the efficiency of OPS can be attributed down to its ability to share data across all nodes within the cluster.

OPS is a true shared data environment because each node is effectively capable of accessing the same cached data at the same time as any other node. Because of this, any disk update operations must be carefully coordinated so that all nodes see the same data in a consistent way. Any OPS instance intending to update or modify data held in a cached data block must first enter into a dialog with the Integrated Distributed Lock Manager (IDLM) to secure exclusive rights to that data. If the data has already been locked because another node is using it, then the IDLM will revoke the request for data rights and force the node to write data to disk where all nodes can see the changes.

It is the IDLM component that allows the I/O to process data so effectively by providing a locking mechanism for Oracle resources that are being used.

2.5.3 Instances synchronization

With OPS, we must have a dedicated link between the different nodes for the synchronization of data. For that, we need to use the IDLM component. The role of IDLM is to coordinate the simultaneous access to shared databases in a way that maintains consistency and data integrity.

2.5.3.1 Integrated distributed lock manager (IDLM) specifications

The aim of IDLM is to coordinate access to the resource by the different instances. It doesn't control the direct access to the tables or objects in the database. All the processes that need access to a resource protected by the

IDLM must open a lock on the resource. Instances access blocks in a per-block level. When an instance locks a block in an exclusive mode, other instances cannot access it. That happens each time an instance modifies a block. In an OPS solution, we have multiple memories and thus, multiple copies of the same data block in each instance's memory. It is the IDLM that ensures the integrity of all these copies. The IDLM tracks all lock requests. If the resource is available the lock is granted; if it is not available, the request is tracked and the access right is granted when the resource is available.

The IDLM maintains a database of the different locks. This database, which is in volatile memory, is distributed among the different nodes or instances, but one node, the master node, usually manages all relevant information about resources and locks queries.

LMON and LMDN processes are used by IDLM. To determine which instance is active, IDLM uses, Cluster Manager (CM) a component of the OSD layer. For further details about CM see 2.5.2.2, "OSD layer components" on page 25.

2.5.3.2 Parallel cache management

The parallel cache management ensures that all SGAs have the same copy of a data block. This is known as a master copy data block in an SGA. The other copies must be identical. The most recent copy of the block exists in an SGA contains all changes made to that block by all instances regardless if any transactions on those instances have committed.

To manage access to shared resources, Oracle uses instance locks called Parallel Cache Management locks (PCM locks). PCM locks ensure cache coherency by forcing requesting instances to have a lock before modifying database blocks.

Non-PCM locks exist; these are row-level locks (for commit or rollback), table locks (for insert, delete, drop table...), system locks, dictionary locks, and library cache locks.

2.5.3.3 Cache fusion

When an instance requests for a block that has been modified by another instance, the block has to be transferred between instances, that is a ping.

In the previous releases of Oracle, the block had to be written on the disk, before the block can be read by another instance. In Oracle8*i*, a new mechanism appeared, the Cache Fusion. This mechanism avoids the need to write the block on the disk, it is transmitted to the requesting instance through the interconnect link.

Cache fusion reduces the access time for a remotely cached block.

The speedup afforded by Cache Fusion enables OLTP and Web applications not designed for OPS to be more easily deployed on a highly scalable, highly available architecture.

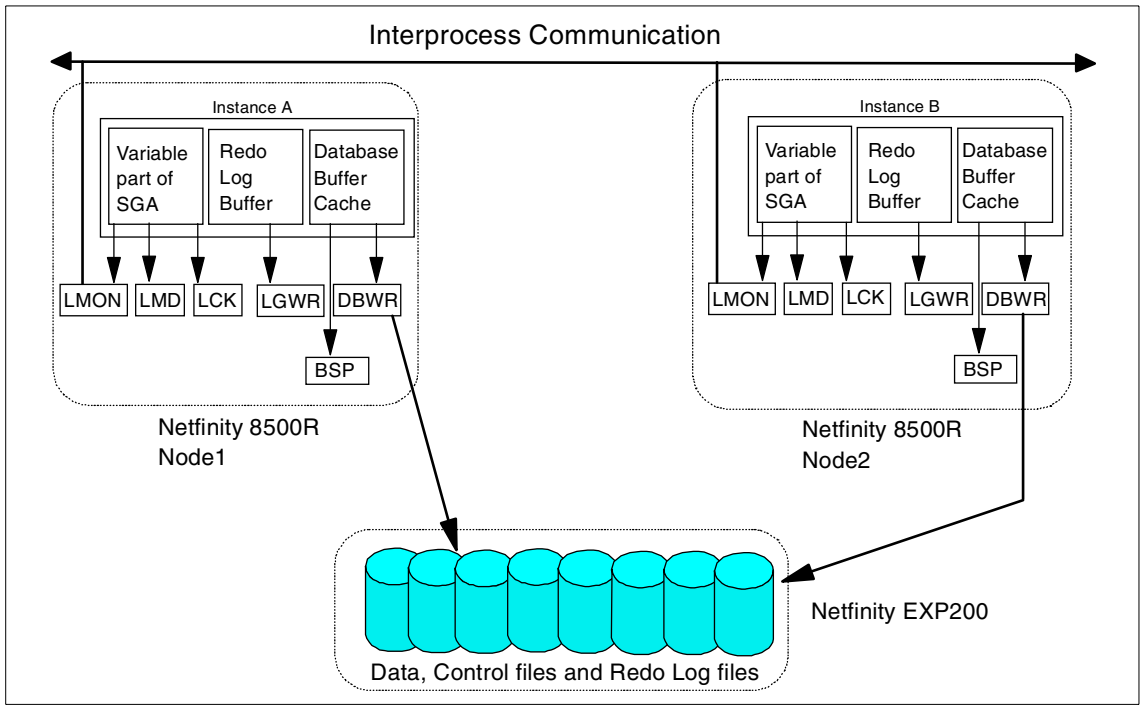


Figure 5. An example of OPS components in a two-node configuration

Chapter 3. Planning for Oracle Parallel Server

In this chapter, we discuss the different considerations that will help you to choose, size and install your Oracle Parallel Server configuration. We describe which components you need to implement OPS, both hardware and software.

3.1 Hardware components

In this section, we will discuss the different hardware components of the IBM Oracle Parallel Server solution. Then, we will review the software components in 3.2, “Software components” on page 32.

The following is an overview of the common hardware components needed for our OPS configuration.

3.1.1 Netfinity Rack

The OPS solutions are only supported as rack solutions. The rack used is the IBM 9306-900 or the 9308-42P NF Enterprise Rack - 42U. The number of racks depends on your configuration - two, four, six, or eight nodes. However, IBM presently supports an OPS two-node configuration running Windows NT 4.0.

For more details, go to the Web site:

`http://www.pc.ibm.com/us/netfinity/storesvr.html#nrs`

3.1.2 Netfinity 8500R

To implement an OPS solution you must have two Netfinity 8500Rs. It is recommended that all the servers have the same components and the same version levels. The Netfinity 8500R configuration should consist of the following components:

- Four CPUs (Pentium III 700MHz with 1MB or 2MB cache)
- Minimum of 4 GB Memory
- Redundant power supply (two power supplies already included)
- 1 MB video memory

3.1.3 Netfinity ServeRAID 3L, 3HB, and 4H Ultra160 adapters

To manage the internal hot swap HDDs you should use a ServeRAID 3L or a ServeRAID 3H controller:

3.1.4 Network adapters

In the configuration you will have multiple network adapters installed in the Netfinity 8500Rs. For the interconnect or private network you should use one 10/100 Etherjet PCI adapter in each server. For maximum performance, these adapters should be set to 100 Mbps and full duplex.

For the public network you can use all supported network adapters.

3.1.5 Netfinity Fibre Channel solutions

The following is an overview of the Netfinity Fibre Channel components that are needed for your OPS configuration.

- Netfinity Fibre Channel Adapter

  Two Netfinity Fibre Channel adapters must be installed in each server. The adapters are used for connectivity between the Netfinity 8500Rs and the Netfinity Fibre Channel Hubs:

- Netfinity Fibre Channel Hub

  Two Netfinity Fibre Channel Hubs are needed for our two node configuration. Two Netfinity Fibre Channel Hubs are used to provide a redundant hardware configuration.

- Netfinity Fibre Channel Controller Unit

  The Netfinity Fibre Channel controller unit is hosting the Hot-Swap RAID Controllers. The purpose of the unit is to manage the external shared disks in the Netfinity EXP200 External Storage Expansion Unit. In most configurations one Netfinity Fibre Channel Controller Unit is sufficient.

  The Netfinity Fibre Channel Controller Unit comes standard with one Fibre Channel RAID controller, also referred to as a *blade*. The supported configuration requires that you install the optional Netfinity Fibre Channel Failsafe RAID Controller.

- Netfinity EXP200 External Storage Expansion Unit

  The Netfinity EXP200 External Storage Expansion Unit contains your external drives. The EXP200 holds up to 10 HDD SCSI disk drives. You can have up to six EXP200s connected to one Netfinity Fibre Channel Controller Unit.

3.2 Software components

The IBM and OPS solution requires each Oracle8*i* node to have the following software components:

- Windows 2000 Advanced Server
- Netfinity Fibre Channel Storage Manager 7.01
- Oracle8*i* Enterprise Edition 8.1.6
- Oracle Parallel Server v8*i*
- IBM Netfinity Advanced Cluster Enabler for OPS V2.1

Note: Use Netfinity FAStT Storage Manager for FAStT configurations.

In the following sections is a review of the actual firmware and drivers required for the Netfinity Fibre Channel solution and the actual drivers we used for our configuration.

3.2.1 Netfinity Fibre Channel Storage Manager 7.01

Listed is the Storage Manager software required for Windows 2000:

- Netfinity Fibre Channel Storage Manager 7.01 for Windows 2000 (Complete Package)
- Netfinity Fibre Channel Storage Manager 7.01 Agent for Windows 2000
- Netfinity Fibre Channel Storage Manager 7.01 Client for Windows 2000
- Netfinity Fibre Channel Storage Manager 7.01 RDAC for Windows 2000
- Netfinity Fibre Channel Storage Manager 7.01 for Windows 2000 README

Common Files:

- Netfinity Fibre Channel Storage Manager 7.01 Migrate
- Netfinity Fibre Channel Storage Manager 7.01 Scripts
- Netfinity Fibre Channel Storage Manager 7.01 SNMP MIB File

3.2.2 Netfinity Fibre Channel Controller firmware and NVSRAM

If necessary to update your Fibre Channel controllers, use the following updates via SM7 Migrate or SM7 Client:

- FW_04000100_04000100.dlp
- NV4766WNT856004.dlp (for 3526 controllers only)
- NV4774WNT856003.dlp (for 3552 controllers)

For updates via the serial interface:

- 0401000100.apd
- 0401000100.bwd

- NV4766WNT856004.dl (for 3526 controllers only)
- NV4774WNT856003.dl (for 3552 controllers only)

In our configuration, we updated the Netfinity Fibre Channel 3526 controllers via the serial interface.

3.2.3 Netfinity Fibre Channel Adapter driver

In our Netfinity Fibre Channel configuration, we used the following adapter driver:

Netfinity Fibre Channel Adapter Driver 07.04.04.02 for Windows 2000

3.2.4 Netfinity Fibre Channel Adapter firmware/NVSRAM

In our Netfinity Fibre Channel configuration, we used the following adapter firmware:

Netfinity Fibre Channel Adapter BIOS version 1.37a

3.3 Netfinity Fibre Array Storage Technology (FAStT) solution

IBM's Netfinity Fibre Array Storage Technology (FAStT) solutions are designed to support the large and growing data storage requirements of business-critical applications. From basic high-speed configurations offering continuous data access and protection, to long distance, high-availability and fully redundant configurations for critical disaster-recovery capabilities, FAStT solutions help meet enterprise storage requirements today and can become the foundation for a SAN tomorrow.

Although we did not use the FAStT solution in our lab configuration, it is important that we mention the availability of this solution and its features and benefits.

The following are the main components of the Netfinity Fibre Array Storage Technology needed for your OPS configuration.

- FAStT500 RAID Controller Unit

  The FAStT500 RAID Controller Unit provides built-in failover protection with no single point of failure. Two hosts and two drive mini-hubs are included with four additional slots available for optional mini-hubs.

  The supported RAID levels are 0, 1, 3, 5, 10.

3.3.1 FAStT500 EXP500 External Expansion Unit

The FAStT500 EXP500 External Expansion Unit provides full Fibre Channel support. It supports up to 10 high-speed Fibre Channel hard disk drives (9.1 GB, 18.2 GB, and 36.4 GB).

3.3.2 FAStT500 Mini-Hub

The FAStT500 Mini-hub provides two GBIC slots to support additional drive enclosures or host systems. It is GBIC-based for hot-plug connectivity.

3.3.3 FAStT Host Adapter

The FAStT Host Adapter provides 100MBps speed and full Fibre Channel fabric support ActiveTM PCI.

3.3.4 Netfinity FAStT Storage Manager

The Storage Manager supports eight independent hosts or eight two-node clusters. The Storage Manager is Java-based to manage FAStT components from single or multiple locations.

For more information regarding Netfinity FAStT visit the following Web site:

`http://www.pc.ibm.com/ww/netfinity/fibrechannel/`

3.3.5 Netfinity FAStT Fibre Channel Adapter drivers

If you use the FAStT Fibre Channel configuration, you should use the following adapter driver:

Netfinity FAStT Host Adapter Device Driver for Windows 2000 Version 1.00

For more information on IBM Netfinity Fibre Channel documentation and downloadable files, go to the Web site: `http://www.pc.ibm.com/us/netfinity`

To locate the necessary firmware, BIOS, drivers and hardware documentation regarding Netfinity Fibre Channel technology, perform the following steps:

1. In the left panel, select **Support**.
2. Under Select your Product, Select **Servers**.
3. Under selection number 2 (Family), select **Fibre Channel Solutions**.
4. In the left panel window, select **Downloadable Files** or **Online Publications**.

5. If you selected Online Publications, under the Online Publications by Category, select **Fibre**.

Chapter 4. Netfinity and Windows 2000 installation

In this redbook, we configured and ran failover tests using Oracle Parallel Server 8*i* R2 (8.1.6) and Windows 2000 Advanced Server running on a two-node IBM Netfinity 8500R cluster (eight processors each) with Fibre Channel architecture and an EXP200 External Storage Expansion Unit. The Netfinity Fibre Channel Storage Manager 7.01 was used to configure the drives in our EXP200.

In Figure 6 is a diagram of the hardware configuration that was implemented in the ITSO lab prior to loading the latest BIOS, firmware, network operating system, drivers, and solution software.

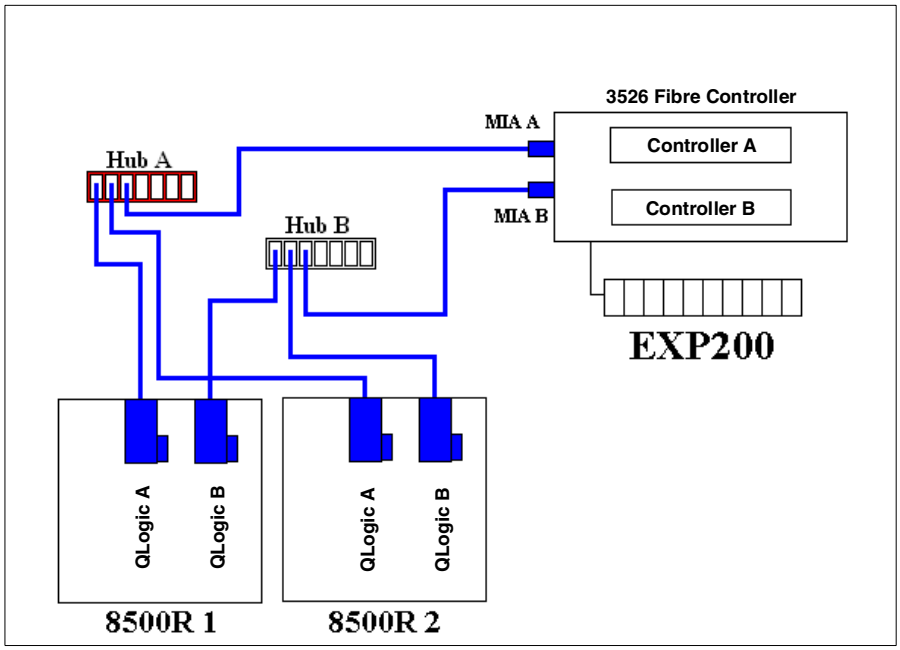


Figure 6. Hardware diagram of Oracle Parallel Server solution at ITSO

4.1 BIOS, firmware, and drivers

It is important that all the different devices in the configuration have the correct microcode level. Now we will go through the different BIOS levels required for the different components of an OPS configuration.

4.1.1 Netfinity 8500R

For the Netfinity 8500R-5RY (PIII - 550MHz) you must have the following levels:

- BIOS: Revision 5 or higher
- Diagnostic:

  To obtain this code, visit the IBM Support site:
  `http://www.pc.ibm.com/support`

4.1.2 Netfinity ServeRAID 3H

The ServeRAID adapter in this configuration was used to configure the internal drives of the 8500R.

For the Netfinity ServeRAID adapters you must have the following levels:

- BIOS: 4.00 or higher
- Firmware: 3.70 or higher
- IBM ServeRAID Device Driver for Windows NT: 4.00 or higher

  Note: These BIOS, firmware, and driver levels apply to ServeRAID 3L and 4H as well.

  To obtain this code, visit the IBM Support site:
  `http://www.pc.ibm.com/support`

4.1.3 Advanced System Management device driver

For the Advanced System Management adapter, you must have the following level:

Advanced System Management Device Driver: 1.16 or higher

To obtain this code, visit the IBM Support site:
`http://www.pc.ibm.com/support`

4.1.4 Fibre Channel Host Adapter

Before installing each adapter in the server, write down the serial number of each Fibre Channel PCI adapter in the event that the non-volatile random access memory (NVRAM) is damaged. If you upgrade the NVRAM or if it is damaged, you will be prompted for the PCI adapter serial number. Install the two Fibre Channel PCI adapters in a 64-bit slot, Bus A slot 1 to 5, if possible.

Version levels:

BIOS: 1.37a or higher

Firmware: 1.37a or higher
Windows 2000 Device Driver: 7.4.3.0 or higher

Important

Each adapter must have the same version of BIOS and firmware.

To obtain this code, visit the IBM Support site: `http://www.pc.ibm.com/support`

4.1.4.1 Update the BIOS and the NVRAM

First create a bootable floppy disk and then copy the update files onto it.

For the BIOS and NVRAM, run the file QL2XUTIL.EXE to flash all of the adapters. From a command prompt run `C:> ql2xutil /L /F`.

When you reboot your system, you will see that each adapter has the same BIOS level.

The Netfinity Fibre Channel PCI Adapter can be configured using the Fast!UTIL utility program. You can access it by pressing ALT + Q during the adapter BIOS initialization. If you have multiple adapters, you will be prompted to choose which adapter you wish to configure.

With the Qlogic Fast!UTIL V1.64 the first option menu is:

```
Configuration Settings
Scan Fibre Device
Fibre Disk Utility
Select Host Adapter
Exit Fast!UTIL
```

From the Configuration Settings menu, select **Host Adapter Settings**. Verify that Host Adapter BIOS is disabled. This is the default value and disables the Fibre Channel disks from being used as boot devices. The internal drives in the server will be used for booting. For more details see Table 1:

Table 1. Fibre Channel Host Adapter and Advanced Adapter Settings

Setting	Options	Default
Host adapter BIOS	Enabled or Disabled	Disabled
Enable LUNs	Yes or No	Yes
Frame size	512, 1024, 2048	2048
Loop Reset Delay	0–15 seconds	8 seconds
Execution Throttle	1 - 256	256
Fast Command Posting	Enabled or Disabled	Enabled
LUNs per Target	0, 8, 16, 32, 64, 128, 256	8
Enable LIP Reset	Yes or No	No
Enable LIP Full Login	Yes or No	Yes
Enable Target Reset	0-255	30
Login Retry Count	0-255	30
Port Down Retry Count	0–255	30
Drivers Load RISC code	Enabled or Disabled	Enabled
Enable Database Updates	Yes or No	No
IOCB Allocation	1–512 buffers	256 buffers
Extended Error Logging	Enabled or Disabled	Disabled

4.1.5 Netfinity Fibre Channel RAID Controller Unit

The Netfinity Fibre Channel Controller Unit must have the following levels:

Bootware: 04.00.01.00 or higher
Appware: 04.00.01.00 or higher
Netfinity Fibre Channel Storage Manager 7.01 or higher

For firmware and NVSRAM updates via SM7 Migrate or SM7 Client:

FW_04000100_04000100.dlp
NV4766WNT856004.dlp (for 3526 Controllers only)

Note: Use NV4774WNT856003.dlp when using 3552 Controllers

For firmware and NVSRAM updates via the Serial Interface

0401000100.apd
0401000100.bwd
NV4766WNT856004.dl (for 3526 Controllers only)

Note: Use NV4774WNT856003.dl when using 3552 controllers

To obtain this code, visit the IBM Support site:
`http://www.pc.ibm.com/support`

Note: If your system is currently configured for Symplicity Manager 6.22, you should migrate your system to Netfinity Fibre Channel Storage Subsystem 7.01 using the SM7 Migrate instructions.

4.1.6 Public network and private network adapters

For the public network we used a 10/100 Ethernet PCI adapter. For the private network we used a 10/100 Etherjet PCI adapter. For the private network (or interconnect), in the setup utility of the 10/100 Ethernet PCI adapter be sure it is set to 100 Mb and full duplex. Windows 2000 automatically recognizes these devices and loads the drivers.

To obtain this code, visit the IBM Support site: `http://www.pc.ibm.com/support`

4.1.7 IBM Hard Drive Update Program Version 1.07

The IBM Hard Drive Update Program Version 1.07 contains hard disk drive firmware updates. IBM PC Servers, Netfinity Servers and IBM Intellistations with specific SCSI hard disk drive options installed may be affected.

IBM strongly recommends this update to all customers to prevent premature failure of drives.

To obtain this code, visit the IBM Support site: `http://www.pc.ibm.com/support`

4.2 Hardware setup

This section is designed to guide the user through the basic installation and configuration of hardware and software prior to the installation of Oracle8*i* Enterprise Edition. It is designed to supplement and not replace the document *Cluster Enabler - Hardware and Software Installation Guide for Oracle Parallel Server* (OPS Installation Guide).

4.2.1 Connecting the Fibre Channel components

Care must be taken when working with the fibre optic cables. Be careful when handling these cables as they are delicate and can be damaged easily. Do not bend, crimp or coil the cables smaller than a 4" diameter. Check how they are installed into the server and cabling arm. Provide enough free cable to allow the server to slide back and forth in the rack.

Attention

Failure to comply with these guidelines is likely to damage the cables and cause signal failure or data loss.

Follow correct cable management procedures, and ensure that your cables are correctly labeled. Visit the the Web site:

`http://www.pc.ibm.com/us/netfinity`

The following instructions will help you locate the necessary documentation regarding Netfinity Fibre Channel technology and cabling:

1. In the left panel, select **Support**.
2. Under Select your Product, select **Servers**.
3. Under selection number 2 (Family), select **Fibre Channel Solutions**.
4. In the left panel window, select **Online Publications**.
5. If you selected Online Publications, under the Online Publications by Category, select **Fibre**.

4.2.1.1 Connecting the IBM Netfinity EXP200

It is likely that in the larger node configuration you will be installing more than one EXP200 External Storage Expansion Unit. It is important that each storage expansion unit attached to the controller is assigned a unique tray ID. This ID is set using the dial at the rear of the module as shown in Figure 7 on page 43.

In our configuration, we used nine 9.1 GB drives in our EXP200.

For further information concerning the EXP200 please visit the Web site: `http://www.pc.ibm.com/us/netfinity/storesvr.html#es` or consult the EXP200 User's Handbook.

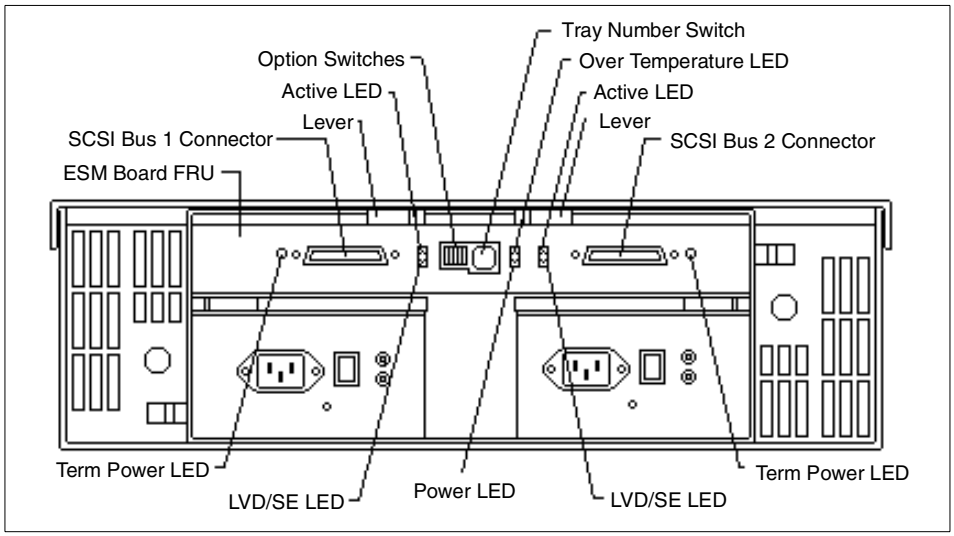


Figure 7. The rear of the EXP200 showing the tray number switch

Read the *IBM Netfinity Fibre Channel RAID Controller Unit User's Handbook* for more detailed cabling instructions.

4.2.1.2 Configure the ServeRAID Adapter

Configure the internal disk subsystem of each node with the ServeRAID configuration utility. This is started by booting from the configuration CD-ROM. Refer to the *IBM Netfinity ServeRAID-3H and 3L Ultra2 SCSI Adapters and User's Guide* for detailed information about using this program.

Although it is feasible for Windows 2000 to boot from the shared storage enclosures, Oracle Parallel Server requires that the operating system and the Oracle program files reside on the non-shared drives within each node. This configuration also offers a higher degree of fault tolerance. However, fault tolerance could be increased further by the addition of a hot spare HDD.

The configuration of the internal disk subsystem remains flexible. However, it is equally important to consider the internal disk subsystem and ensure that it doesn't become the weak link in the high-availability chain.

4.2.2 Windows 2000 installation

This section covers the installation of Microsoft Windows 2000 Advanced Server and IBM Netfinity Fibre Channel Adapter device drivers. Installation of the other components will be covered later in this redbook.

4.2.2.1 Install Microsoft Windows 2000 Advanced Server

Install Microsoft Windows 2000 Advanced Server on each node. Refer to the documentation that is supplied with Windows 2000 Advanced Server for detailed instructions regarding installation procedures.

It is recommended that the servers be installed as member servers or stand-alone servers. In our setup, we installed our servers as member servers belonging to the domain headquarters.world.com. Performance can be hampered by installing as PDC or BDC, since the server will have to use resources to process user requests and authenticate logon requests.

4.2.2.2 Install the Fibre Channel Host Adapter drivers on each node

You will need to install the Fibre Channel Host Adapter drivers on each node. Obtain the latest drivers from the IBM Support Web site:

```
http://www.pc.ibm.com/support.
```

Note that after installing the adapter in the computer system and restarting the system, Windows 2000 will automatically detect the hardware. At this time, you should be prepared to install the correct Fibre Channel adapter driver for your hardware. After inserting the Fibre Channel adapter driver diskette into the A: drive and clicking **OK** to continue, you will see a window similar to Figure 8. Click **Yes** to continue.

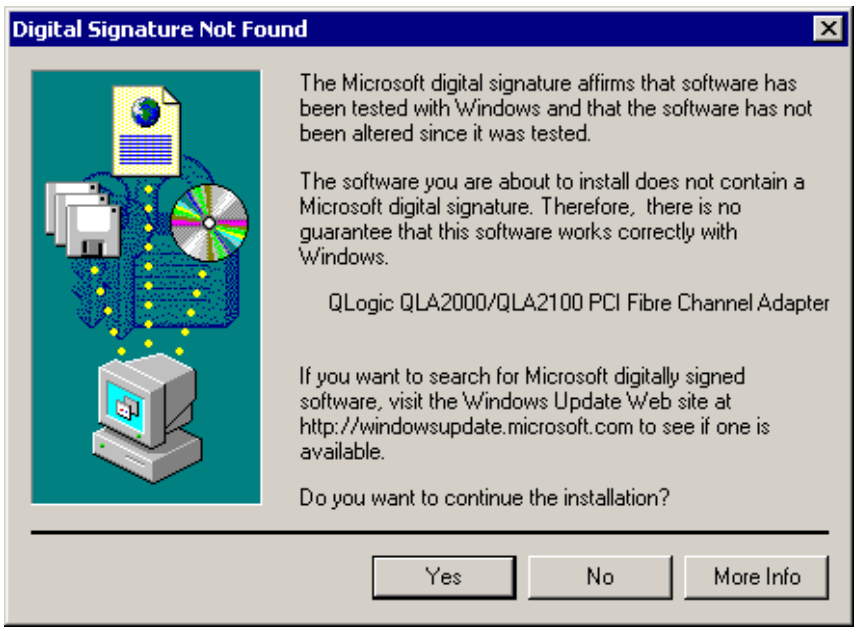


Figure 8. Qlogic PCI FIbre Channel Adapter driver installation window

You can also install the Fibre Channel adapter driver by clicking **Start -> Programs -> Administrative Tools -> Computer Management -> Device Manager**.

4.2.2.3 Modification of the registry

Windows 2000 Advanced Server includes enhanced scatter/gather list support for doing very large SCSI I/O transfers. Perform the steps in the section only if you are using the QL2100 adapters. This step is unnecessary if you are using the QL2200 adapters.

Note: The OEMSETUP.INF file has been updated to automatically update the registry to support 65 scatter/gather segments. Normally, no additional changes will be necessary as this typically results in the best overall performance. If you wish to alter this value, please use the following procedure:

1. Click **Start**, select **Run**, and open the REGEDT32 program.
2. Select **HKEY_LOCAL_MACHINE** and follow the tree structure down to the QLogic driver as follows:

   HKEY_LOCAL_MACHINE

   SYSTEM

   CurrentControlSet

   Services

   QL2100
3. Select the **QI2100** key and use the Edit/Add key to create a new subkey named Parameters.
4. Select the new **Parameters** key and use the Edit/Add key to create a new Device subkey. If multiple adapters are installed, you can specify separate subkeys for each adapter by using a subkey name of DeviceN, where N = the SCSI host adapter number (0, 1, 2, etc.).
5. Select the **Device** key and use the Edit/Add key to add a new value name of MaximumSGList. Set the data type to REG_DWORD and enter a value from 16 to 255 (10 hex to FF hex). A value of 255 (FF hex) enables the maximum 1 MB transfer size. Setting a value higher than 255 results in the default of 64 KB transfers. The default we use is 65 (41 hex).
6. Exit the Registry Editor, then shut down and reboot the system.

Windows 2000 Advanced Server supports the NumberOfRequests registry parameter to specify the maximum number of outstanding requests per

adapter. When the QLogic driver is installed, the registry will be automatically updated with this parameter set to a value of 224.

Attention

Do not increase this parameter above 224. Doing so can result in a system failure.

4.2.3 Video controller resolution

The S3 Inc. Trio3D video controller is a standard device in the Netfinity 8500R. Make sure the display settings for this device is set to 800 x 600 at 256 colors.

Chapter 5. Netfinity Fibre Channel Storage Manager

Netfinity Fibre Channel Storage Manager is software that allows you to configure and manage the Netfinity Fibre Channel RAID Controller Unit and the Netfinity FAStT RAID Controller Unit. The previous version of this software was known as Symplicity Manager 6.22.

In this chapter, we cover the installation of the Netfinity Fibre Channel Storage Manager.

5.1 Components topology

The Netfinity Fibre Channel Storage Manager 7.01 requires the following software and services to be installed on the managing system unit.

- Netfinity Fibre Channel Storage Manager 7.01 RDAC for Windows 2000
- Netfinity Fibre Channel Storage Manager 7.01 Agent for Windows 2000
- Netfinity Fibre Channel Storage Manager 7.01 Client for Windows 2000

  Note: Netfinity Fibre Channel Storage Manager 7.01 RDAC for Windows 2000 should be installed on both cluster nodes.

5.2 Installation

This section describes the installation of the Netfinity Fibre Channel Storage Manager and its components.

5.2.1 Install Netfinity Fibre Channel Storage Manager

Use the Netfinity Fibre Channel Storage Manager CD to install the required software. Execute `Setup.exe` from each respective subdirectory. When installation is complete, restart and ensure that the Disk Array Monitor service has started.

- `x:\Rdac`
- `x:\SM7agent`
- `x:\SM7client`

`x: = CD-ROM drive letter`

5.3 Netfinity Fibre Channel Storage Manager configuration

When the installation of Netfinity Fibre Channel Storage Manager is finished, you can run the Client interface. You will be able to see the controllers in an optimal status and all the drives (see the example in Figure 9).

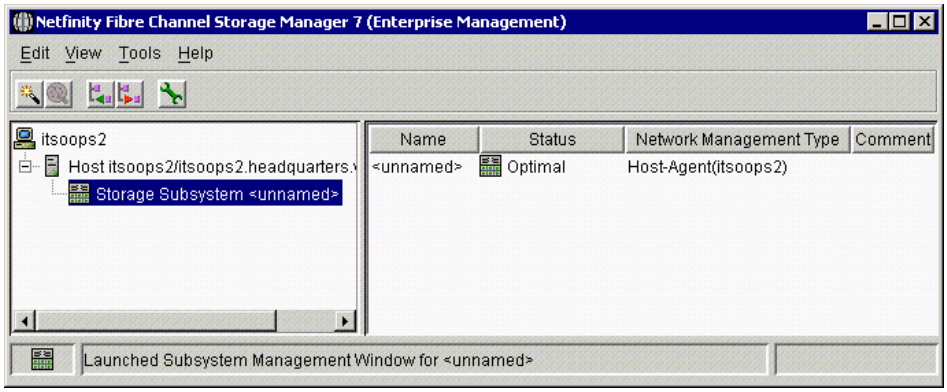


Figure 9. Netfinity Fibre Channel Storage Manager window

From the client interface, you can view the storage subsystem in greater details by double-clicking **Storage Subsystem**. You will see a window similar to Figure 10 on page 49. From this interface, you are able to configure, monitor, and manage the EXP200.

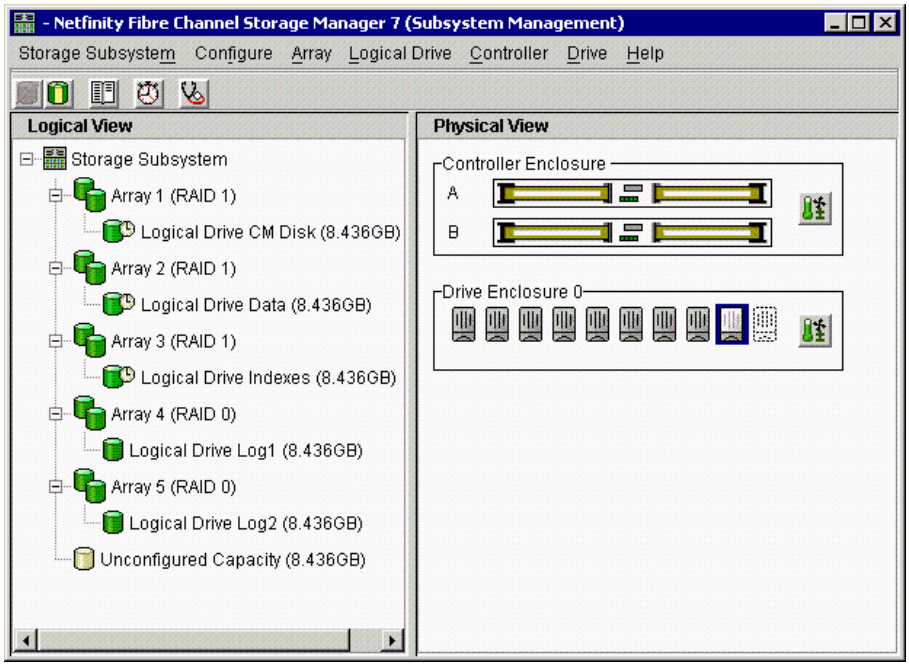


Figure 10. Netfinity Fibre Channel Storage Manager 7 (Subsystem Management) window

We recommend that you configure your controllers as Active-Active. If one of your controllers were configured as Active-Passive (for example, Controller B), the controller would be a shaded grayed color.

We recommend that Controller A support the CM-disk and the index arrays and Controller B support the data and log arrays. This configuration is recommended because of the extensive writing that takes place and we separate the arrays among the two controllers to avoid any possible bottlenecks.

Note: If you execute the Netfinity Fibre Channel Storage Manager and you receive a blank screen, here are some of the possible problems you may have encountered:

- Firmware and driver for Fibre Channel adapter incompatible.
- Fibre Channel cable failure.
- Firmware and NVSRAM for controller needs updating.
- Use serial link and Web instructions.

5.3.1 Verifying controller firmware

If your RAID controller unit does not contain the most recent version of controller firmware, you will need to download the correct version.

The version requirements are:

Bootware: 04.00.01.00 or higher
Appware: 04.00.01.00 or higher
NVSRAM: NV4766WNT856004
Netfinity Fibre Channel Storage Manager 7.01 or higher

You can determine the current version of bootware, firmware and Fibre Channel on the Storage Subsystem Profile window. This can be viewed by selecting **Storage Subsystem -> Profile**.

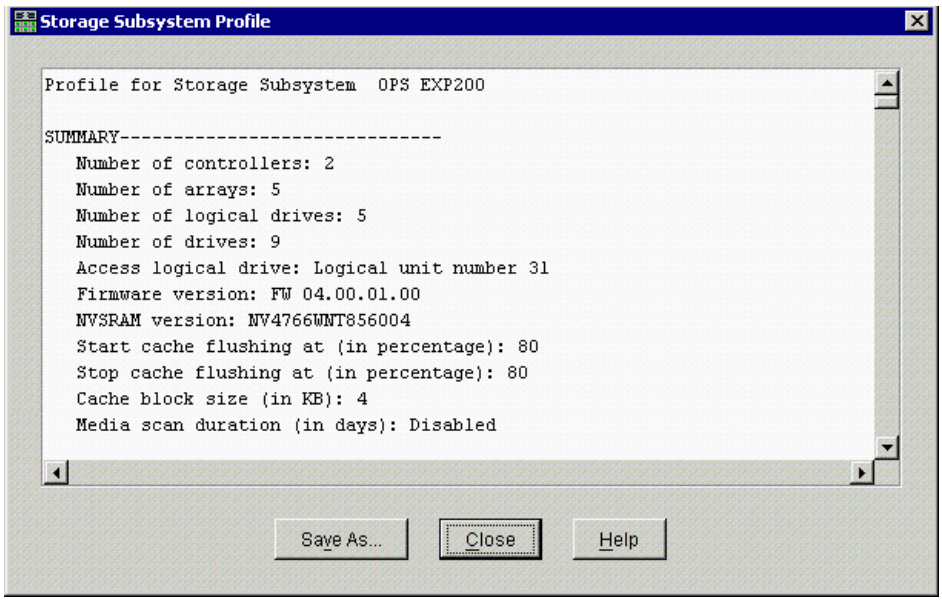


Figure 11. Summary of Storage Subsystem Profile window

If the levels are not the same for the two controllers or they are below the levels required, you should proceed to an upgrade.

Note: If your system is currently configured for Symplicity Manager 6.22, you should migrate your system to Netfinity Fibre Channel Storage Subsystem 7.01 and use the SM7 Migrate instructions.

To download the latest level and migration instructions, go to:
`http://www.pc.ibm.com/support`

Perform the following steps to download the controller firmware:

1. From the IBM support Web site, download the files to the \program files\symsm\lib directory.
2. Download the files to your managing system or create a CD.
3. Go to the Web site: `http://www.pc.ibm.com/us/netfinity`

   The following instructions will help you locate the necessary firmware.

   a. In the left panel, select **Support**.

   b. Under Select your Product, select **Servers**.

   c. Under selection number 2 (Family), select **Fibre Channel Solutions**.

   d. In the left panel window, select **Downloadable Files**.

Shown in Figure 12 and Figure 13 on page 52 are examples of downloading firmware and NVSRAM files from the Fibre Channel Storage Manager.

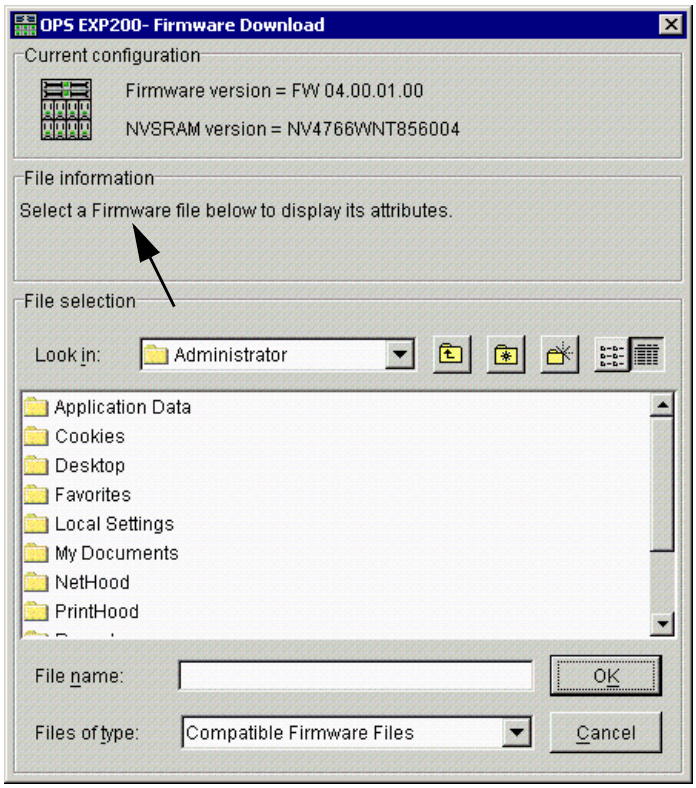


Figure 12. Firmware Download window

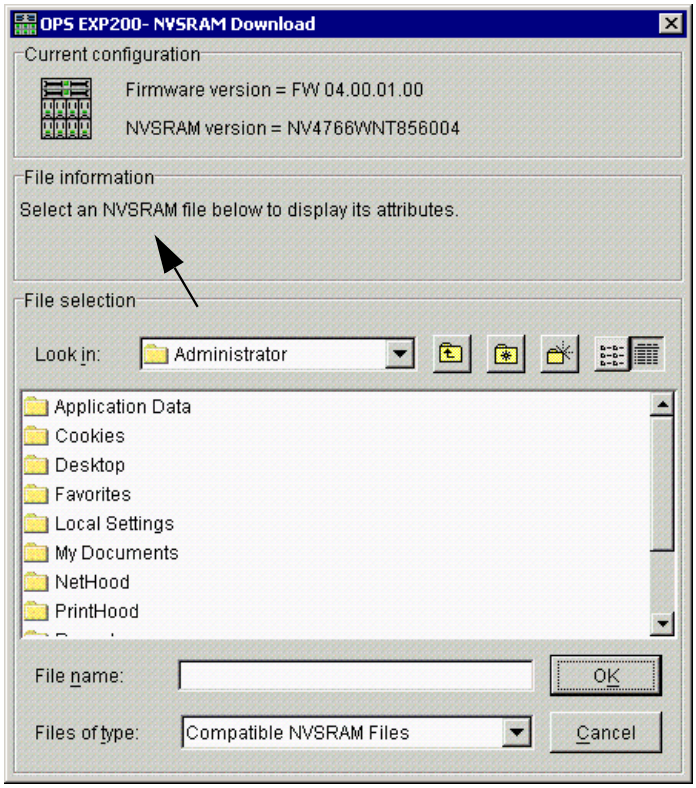


Figure 13. NVSRAM Download window

5.3.2 Redundant controllers

In our configuration, we have two controllers set up as redundant controllers. That means if one controller fails, the other controller takes over the failed controller's functions, and the RAID controller unit continues to operate normally.

The redundant controller feature is managed by the RDAC software, which controls data flow to the controller pairs independently of the operating system. This software keeps track of the current status of the connections and can perform the switchover without any changes in the operating system.

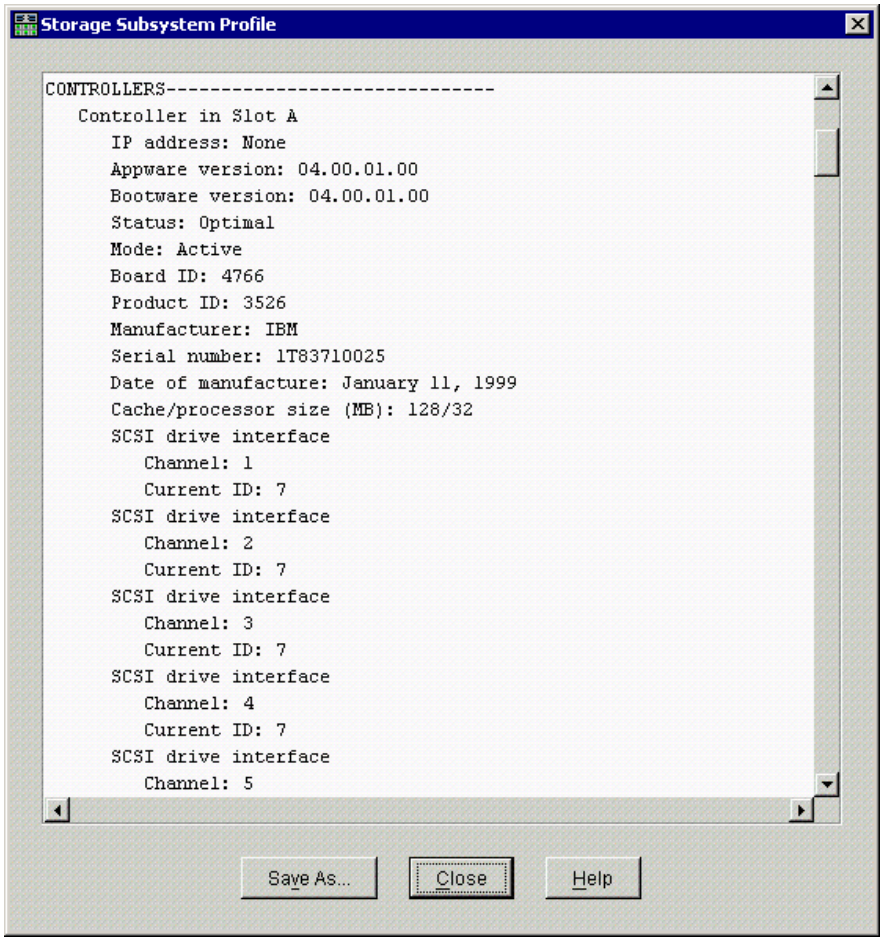


Figure 14. Storage Subsystem Profile window for Controller A

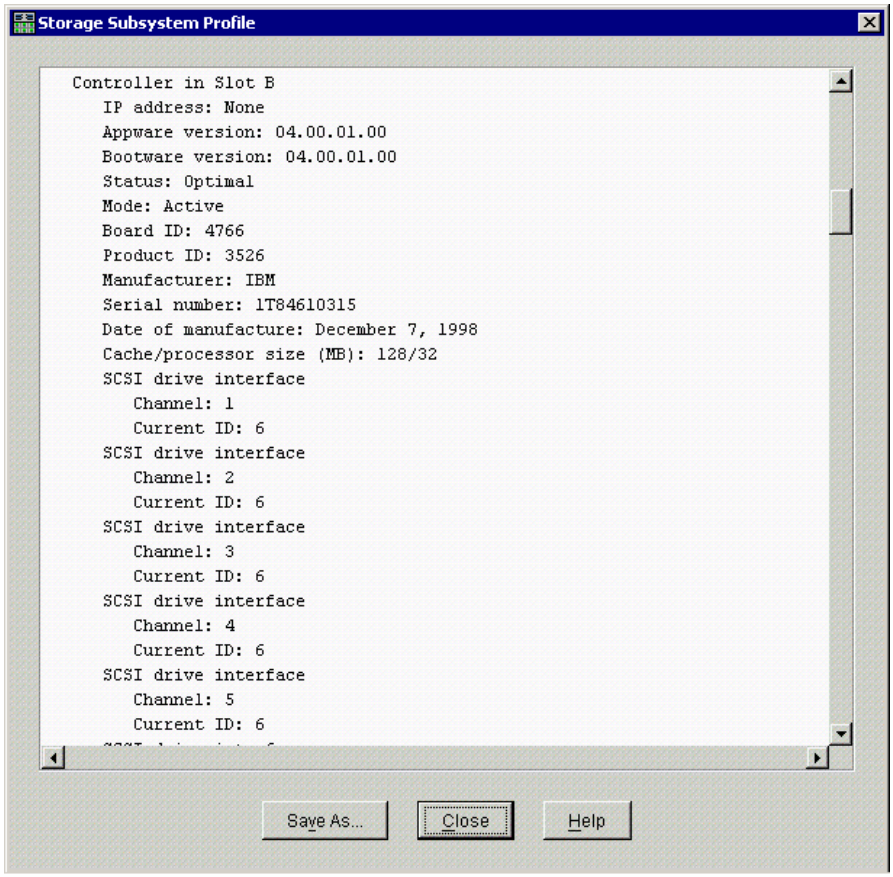


Figure 15. Storage Subsystem Profile window for Controller B

The redundant controllers can be configured in two ways:

- Active/Passive: The second controller is held as a spare, taking over if the first controller fails.
- Active/Active: Each controller handles I/O from the host. It is possible to assign a different LUN to each controller. If one fails, the other one takes over the failed controller's assigned LUNs. With an OPS solution you should configure the controllers Active/Active, because using two controllers for I/O is faster than using a single one.

You need to check if the two controllers are in the correct mode.

For more information on Netfinity Fibre Channel, go to the Web site:

`http://www.pc.ibm.com/us/netfinity`

The following instructions will help you locate the necessary firmware, BIOS, drivers and hardware documentation regarding Netfinity Fibre Channel technology:

1. In the left panel, select **Support**.
2. Under Select your Product, select **Servers**.
3. Under selection number 2 (Family), select **Fibre Channel Solutions**.
4. In the left panel window, select **Downloadable Files** or **Online Publications**.
5. If you selected Online Publications, under the Online Publications by Category, select **Fibre**.

Chapter 6. Operating System Dependent (OSD) layer

This chapter covers the OSD components and the installation of this code. The OSD code used during this installation is the IBM Netfinity Advanced Cluster Enabler for Oracle Parallel Server V2.1.

6.1 Installing the OSD layer

This section will guide you through the installation of the OSD layer code. This should be completed prior to installing Oracle8*i* R2. Contact IBM to obtain the OSD code.

6.1.1 Preparation - all nodes

It is important to ensure that you have completed the following steps prior to installing on *all* nodes:

1. Ensure that each node is up and that you are logged on with administrator privileges on each node.
2. Ensure the hosts file (c:\winNT\system32\drivers\etc\hosts) is mapping:
   - The host name (with and without the domain name) to the TCP/IP address for each node
   - The interconnect host name to the interconnect TCP/IP address for each node

   This needs to be performed on every node. See Figure 16 on page 58.

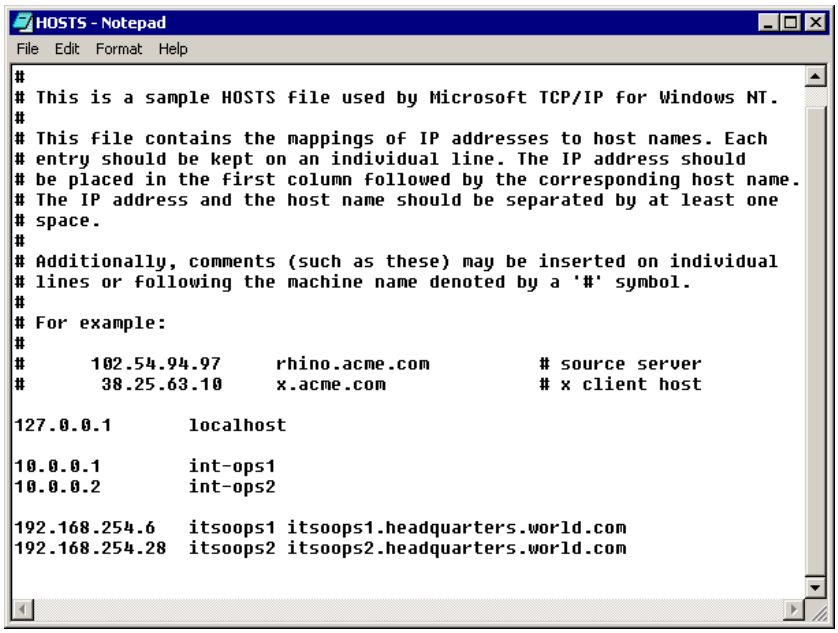


Figure 16. c:\win2000\system32\drivers\etc\hosts window

3. Ensure you have an internal drive with enough space (1 GB) for your OSD and Oracle installation on each node. This internal drive must have the same drive letter on each node. We have chosen the drive D: for our installation.
4. Synchronize the time and date across all nodes using the `net time` command from a command prompt as outlined below:

```
C:\>net time /set \\ITSOOPS1
current time at \\ITSOOPS1 is 3/23/99 7:08 PM

The current local clock is 3/23/99 7:06 PM
Do you want to set the local computer's time to match the time
at \\ITSOOPS1? <Y/N> [Y]: y

The command completed successfully.
```

5. Make registry backups (this will enable you to cleanly start the install of both the OSD and Oracle8*i* if you experience difficulty).

Note: All nodes should be pinged from each node to ensure that network names can correctly resolve the nodes and that the nodes can communicate.

6.1.2 Installation on the first node

Complete the following steps on the first node. Installation will be performed automatically on other nodes:

1. In our example, we obtained IBM Netfinity Advanced Cluster Enabler for OPS V2.1 directly from development and copied the self-extracting file to a directory on the node.

   Note: IBM Netfinity Advanced Cluster Enabler for OPS V2.1 is shipped on a CD-ROM. During a normal installation, you will load the CD-ROM into your CD-ROM drive to begin the install. Review the license agreement and then direct the program where to install the software.

2. Double-click this file. The program will ask you for a location where you want to extract the files. Enter a directory on the drive you have planned to handle the OSD and Oracle directory (for our example, we installed it in D:\Oracle\osdbin, drive D being available on each node). The configuration program will be launched automatically.

3. Using Figure 17 as a guide, enter and add each interconnect node name and click **Add New Node**.

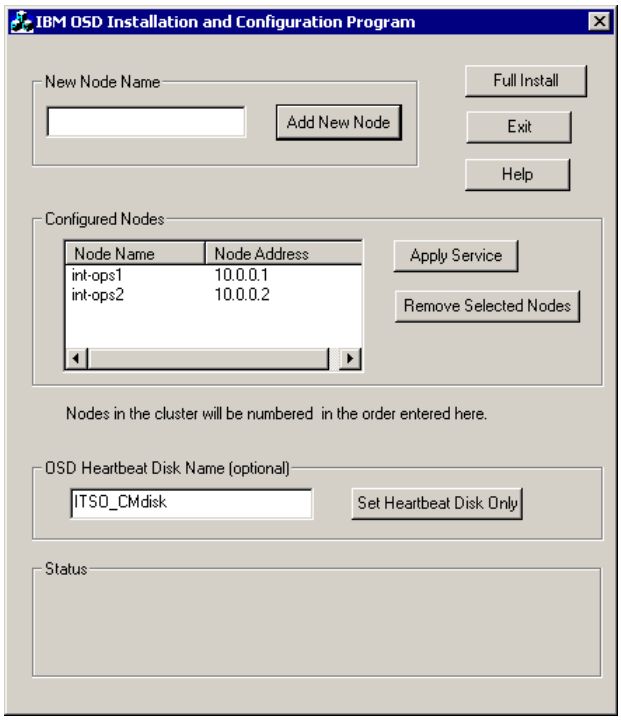


Figure 17. Configuring the OSD window

4. Enter an OSD Heartbeat Disk Name.

   Note: Remember the name you have entered; you will need it in Chapter 8, “Creating and validating a database” on page 77. We have chosen “ITSO_CMdisk”. Usually, it is the database name followed by “_CMdisk”. You will be able to return to this configuration program and change the values by launching the osdconfig.exe program located in the OSD directory (D:\Oracle\osdbin).

5. When all the nodes are listed in the Configured Nodes box, click **Full Install** to install the OSD on all nodes.

6. Verify that you see a message indicating the installation is complete. Click **Exit.**

7. Open Services from the Computer Management window. Ensure that the Oracle CM and NM services have started successfully. Check services for all nodes (see Figure 18).

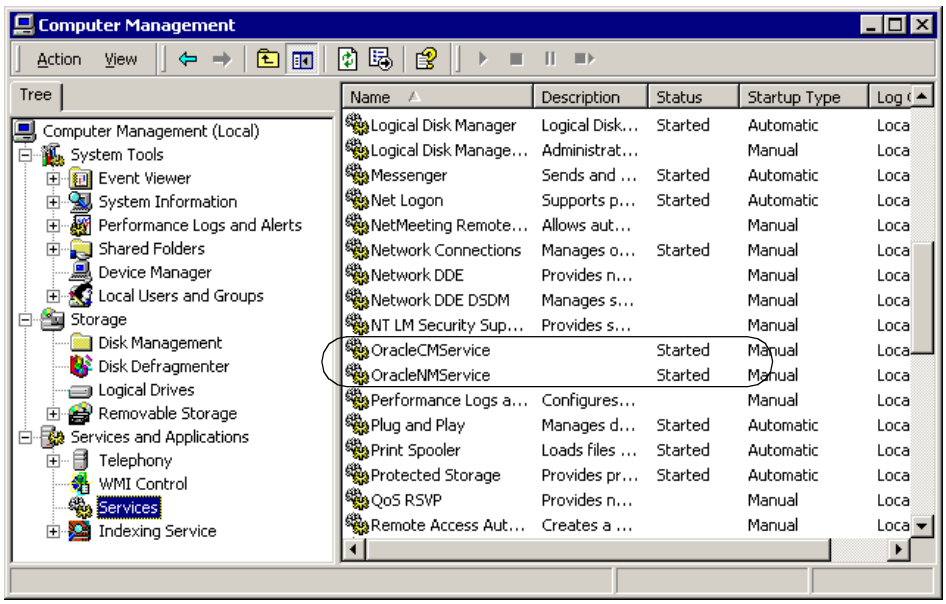


Figure 18. Cluster Management and Node Monitor services on all nodes

8. You do not have to reboot after the OSD installation.

Chapter 7. Installing Oracle8*i* Enterprise Edition

This chapter guides you through the installation of Oracle8*i* R2 with the Parallel server option including the non-cluster aware services in detail.

7.1 Preparation - all nodes

It is important to ensure that you have completed the following steps prior to installing on all nodes:

- Complete the IBM Netfinity Advanced Cluster Enabler for OPS V2.1 installation (see Chapter 6, “Operating System Dependent (OSD) layer” on page 57).
- Ensure that each node is up and that you are logged on as administrator on each node.
- Ensure that OracleCMService and OracleNMService are started on all nodes.
- Make registry backups (this will enable you to cleanly start the installation of Oracle8*i* Enterprise Edition should you experience difficulty).

7.2 Main code installation

Complete the following steps on the first node only; other nodes will be installed automatically:

1. Insert the Oracle 8.1.6 Enterprise Edition CD-ROM. The CD-ROM should autostart.
2. Click **Install/Deinstall Products** and then click **Next.** You will see a window similar to Figure 19.

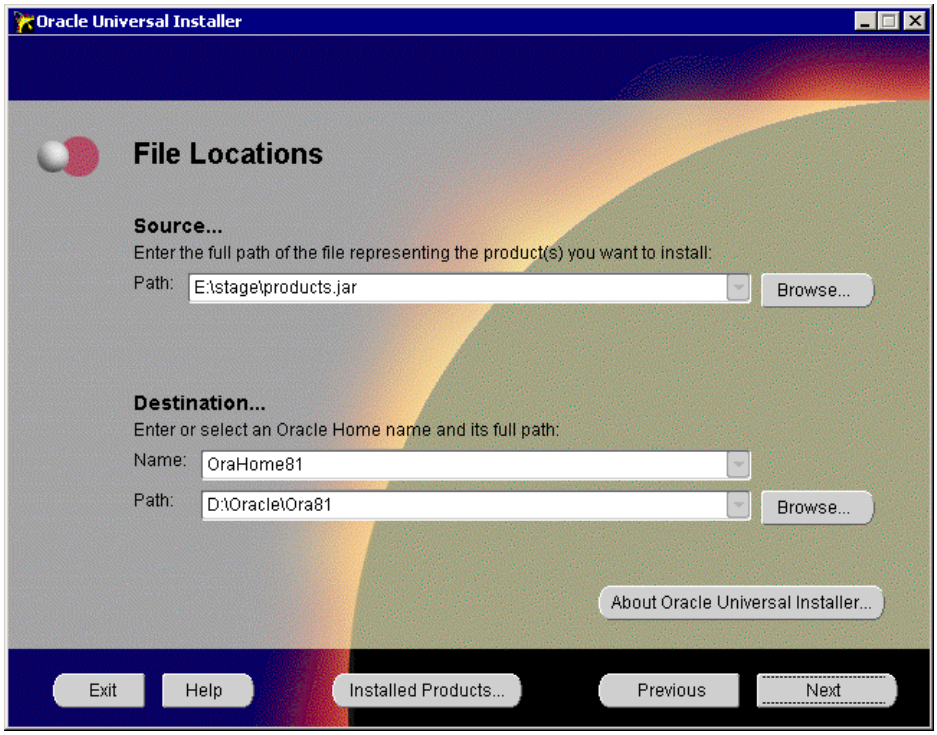


Figure 19. Specifying the location window

3. In the Destination Path, enter the Oracle directory on the drive you have planned to handle the OSD and Oracle directory (in our example, we installed it in D:\Oracle\Ora81, drive D being available on each node). Keep the default for the other fields. Click **Next**. You will see a window similar to Figure 20.

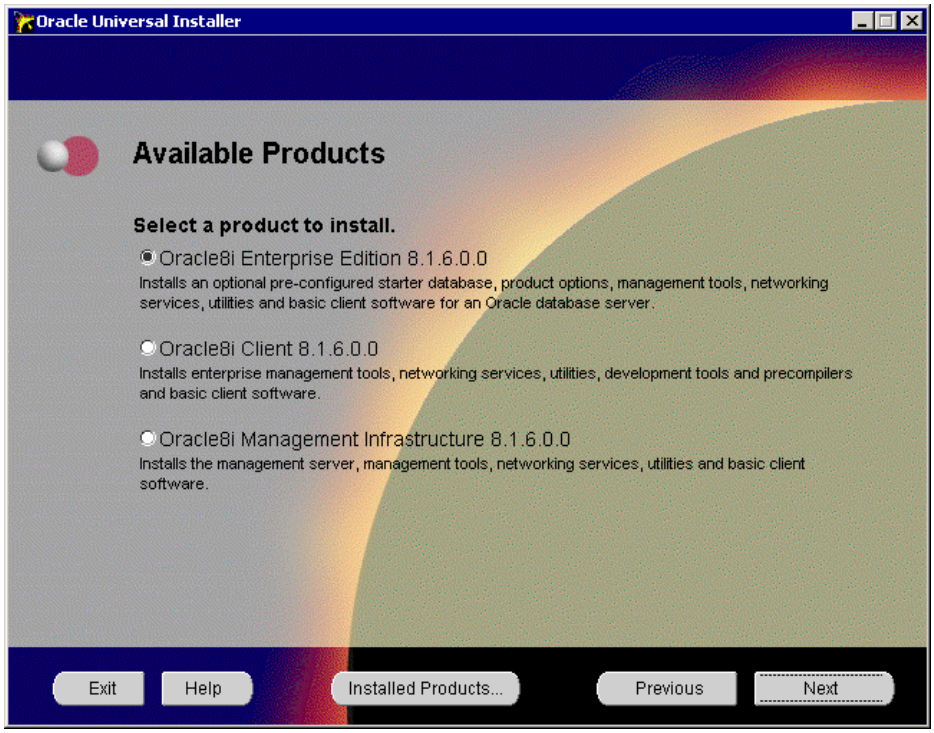


Figure 20. Choosing the product to install

4. From the Available Products screen, select **Oracle8*i* Enterprise Edition 8.1.6.0.0**. Click **Next**. You will see a window similar to Figure 21.

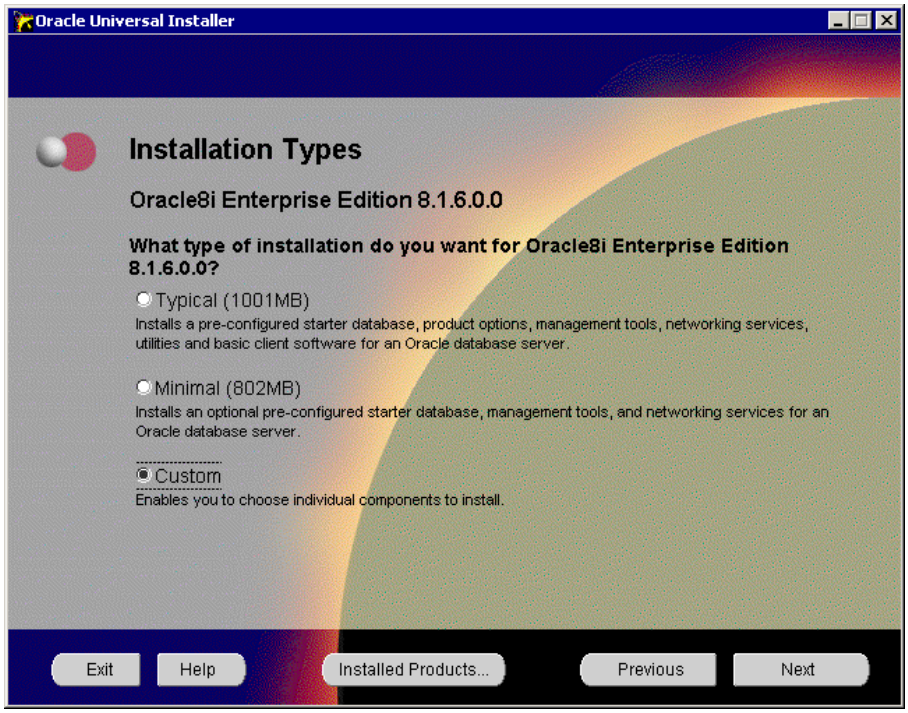


Figure 21. Choosing the Installation Type

5. Select **Custom** installation and click **Next**. You will see a window similar to Figure 22.

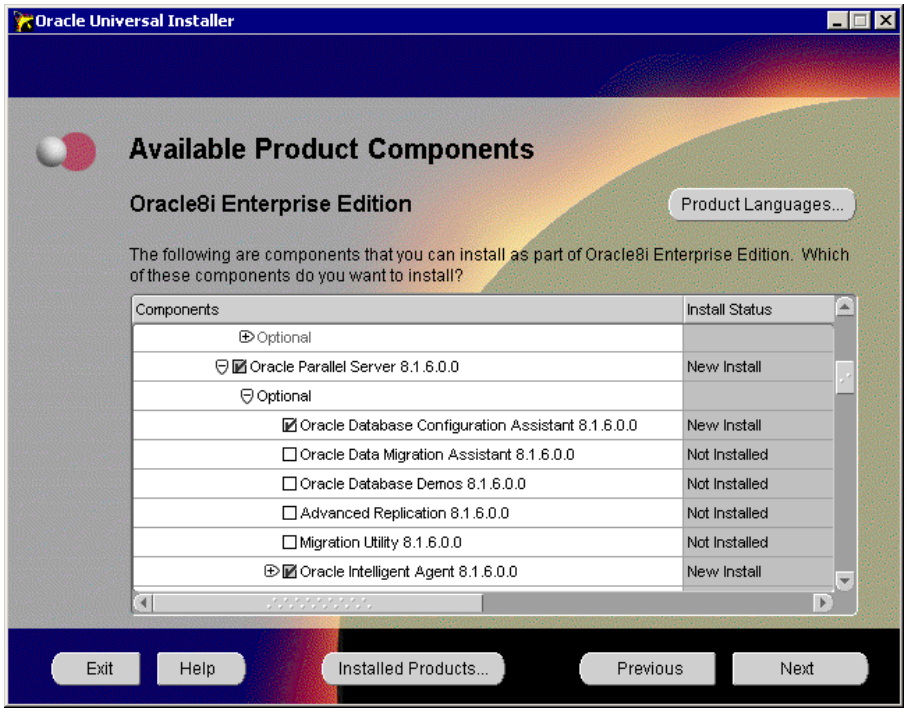


Figure 22. Available Product Components window

6. This option allows you the opportunity to load additional components including OPS. Select and deselect the following components:
   - Select **Oracle Parallel Server 8.1.6.0.0**
   - Select **Oracle Database Configuration Assistant 8.1.6.0.0** (under the Oracle Parallel Server tree)
   - Select **Oracle Intelligent Agent 8.1.6.0.0** (under the Oracle Parallel Server tree)
   - Select **Oracle for Windows NT performance monitor**
   - Select **Universal Installer**
   - Deselect **Oracle Advanced Security** unless you are using an external authentication program such as SecurID or CyberSafe for your database protection
   - Deselect the **Oracle Management Server** unless you plan to use the OPS server as management server, in Chapter 10, “Database migration” on page 141 we are going to install the management server on a separate workstation.

7. Click **Next**. You will see a window similar to Figure 23 on page 66.

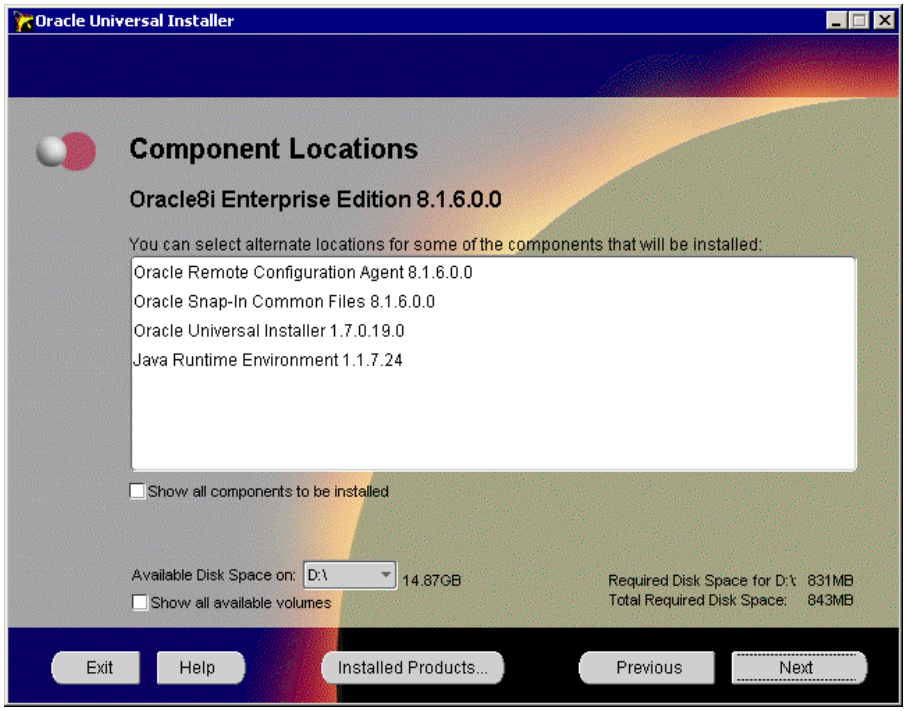


Figure 23. Components Locations window

8. Certain components can be kept in different drives. You will be given the choice of selecting different locations for these components. Click **Next**. You will see a window similar to Figure 24.

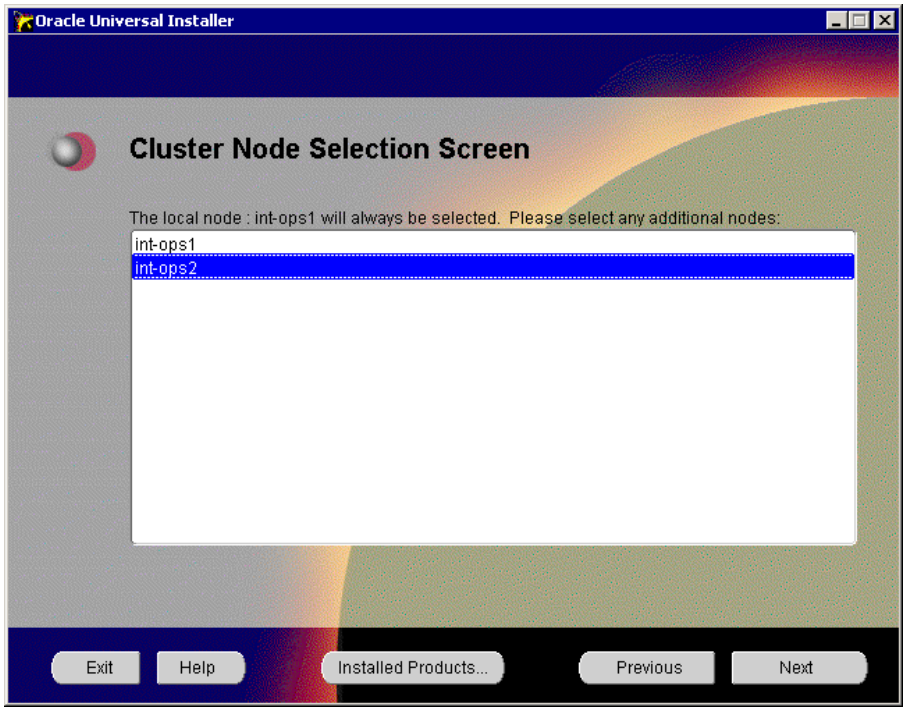


Figure 24. Cluster Node Selection Screen

9. In Figure 24, highlight the nodes on which you will install Oracle8*i* Enterprise Manager and the other selected components.

   Select **int-ops2** and click **Next**.

   Note: The local node, int-ops1 will always be selected.

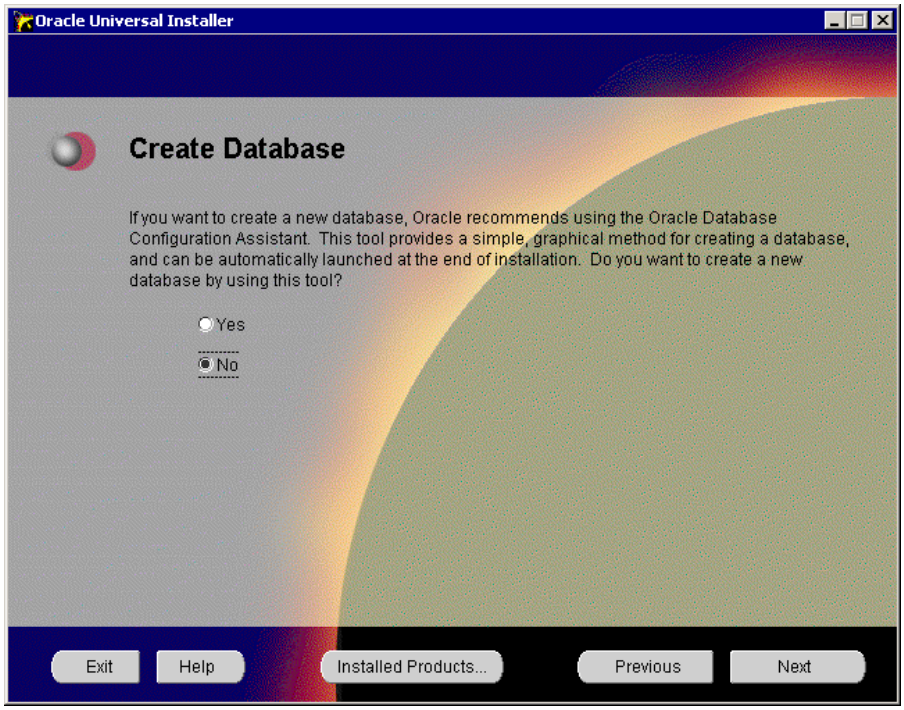


Figure 25. Create Database window

10. You will be prompted to create a database. Select **No** and click **Next**. You will see a window similar to Figure 26.

    Note: The creation of the database is performed in Chapter 8, “Creating and validating a database” on page 77.

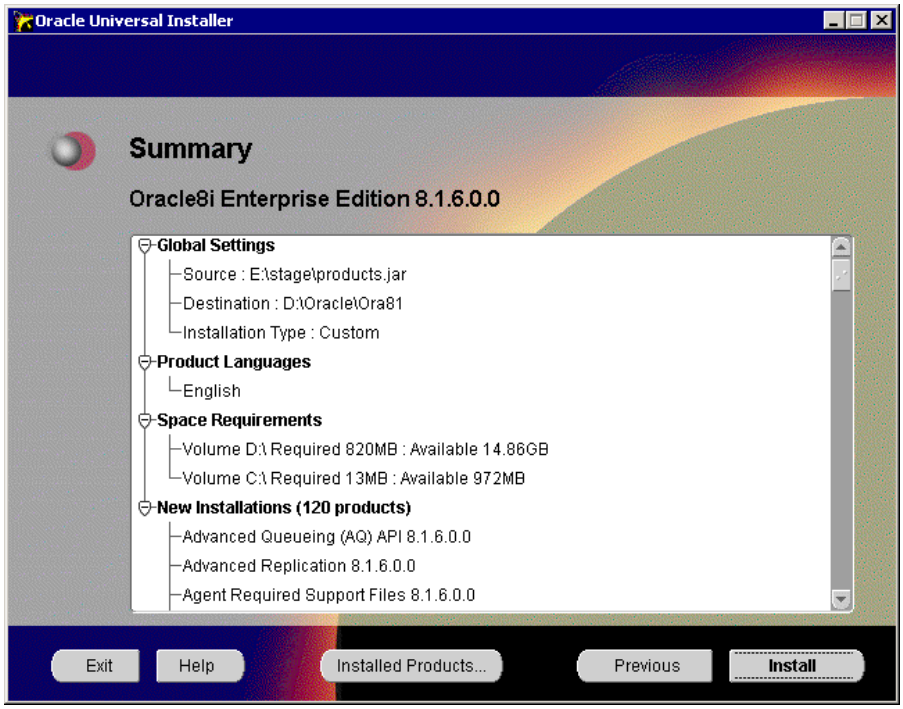


Figure 26. Summary of installation

11. Check the summary page. This will display a tree of products that will be installed. Check the install list. If you need to install additional components, click the **Previous** button. If this list is correct, click **Install**.
12. All code will be installed to the local node first. Code will be installed to subsequent nodes one node at a time. This is carried out automatically and should require no user interaction. An installation log will be created. This is excellent for error determination should there be a problem with the install. The log file can be found at: C:\ProgramFiles\Oracle\Inventory\Logs\installActions.log
13. After the installation bar has reached 100%, wait for disk activity across all nodes to stop before clicking **Next**. The installation may pause for a short time at this point. This is normal. You will be presented with the Net8 Configuration Assistant window (as shown in Figure 27 on page 70).

Note: If you have clicked in the installation window during the installation process, the Net8 Configuration Assistant window will appear at the rear of the installation window. Bring the Net8 Configuration Assistant back to the forefront.

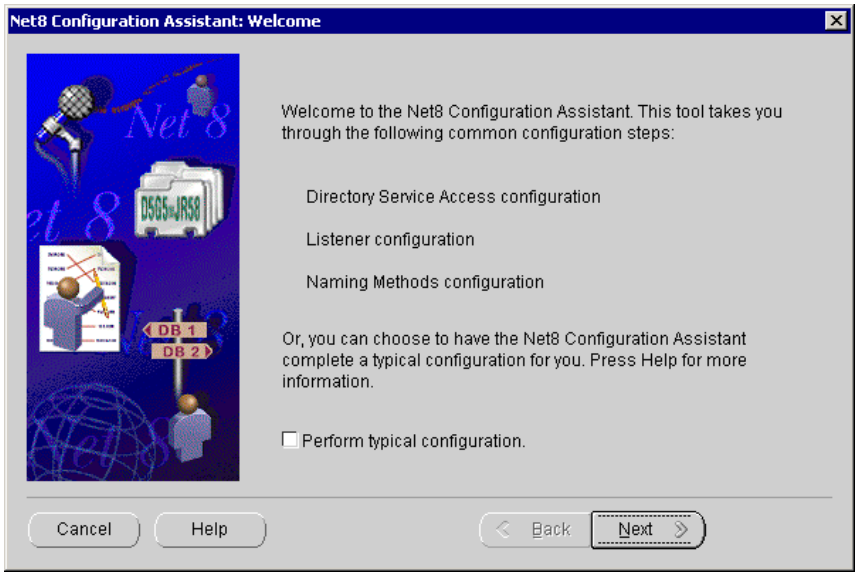


Figure 27. Net8 Configuration Assistant window

This section details the listener configuration window which is a required service to enable remote connection to the database. It also looks at the naming methods configuration and the listener service configuration.

14. At the Net8 Configuration Assistant window (shown in Figure 27), click **Next**.

15. A progress window will appear, detailing the progress of Net8 Configuration, and any other additional components in the install queue. A new window will appear in the foreground to guide you through the listener configuration.

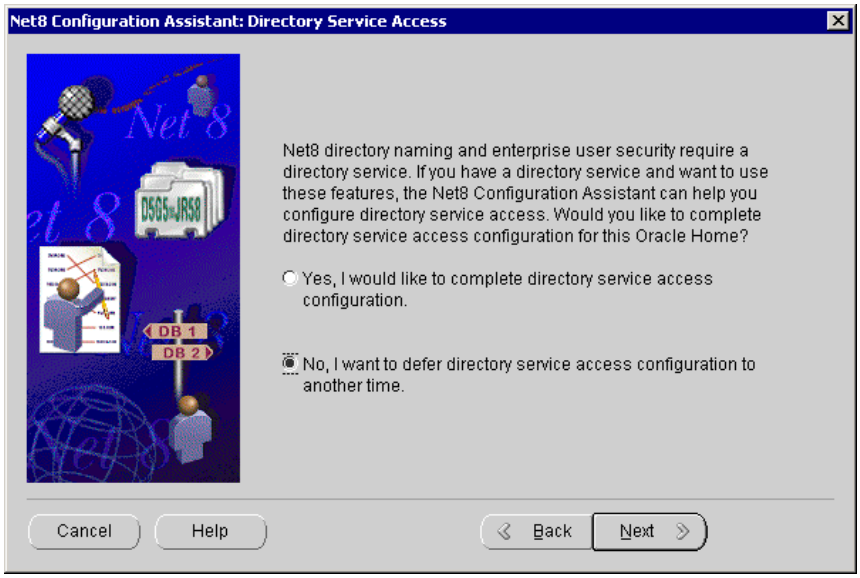


Figure 28. Directory Service Access window

16. You will be prompted to configure the Directory Service Access (Figure 28). Select **No, I want to defer directory service access configuration to another time**. Click **Next**. You will see a window similar to Figure 29.

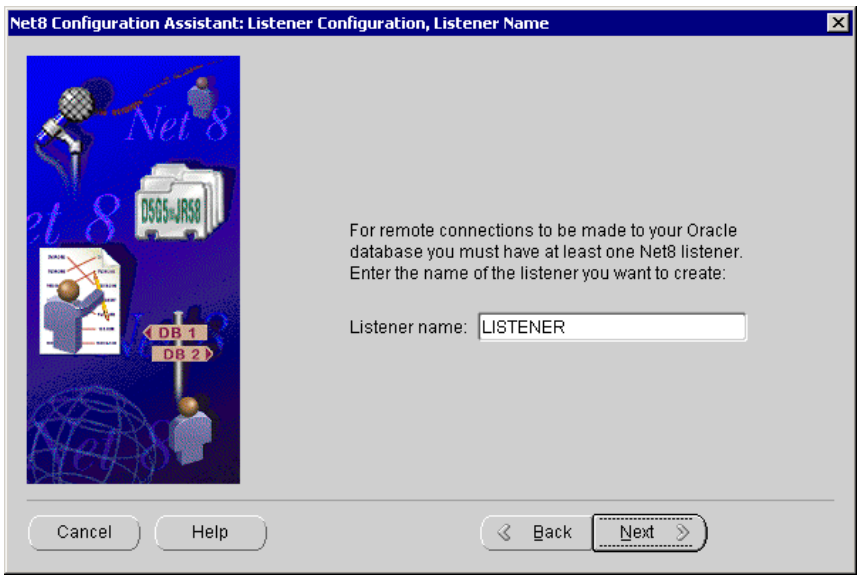


Figure 29. Listener Name window

17. On the Listener Name window (Figure 29), type a listener name or use the default `LISTENER`. Click **Next**. You will see a window similar to Figure 30.

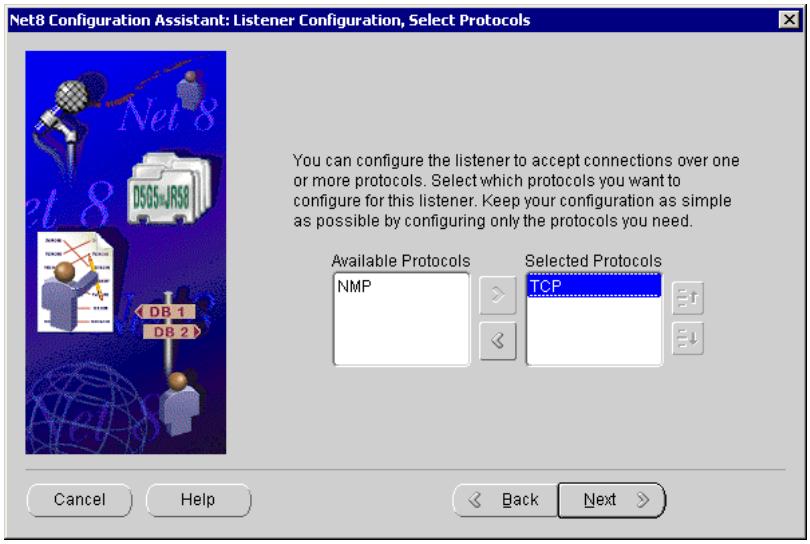


Figure 30. Listener Configuration, Select Protocols window

18. On the Select Protocols window, select **TCP**. Click **Next**.
19. On the TCP Client window, select **Net8 Clients**. Click **Next**.

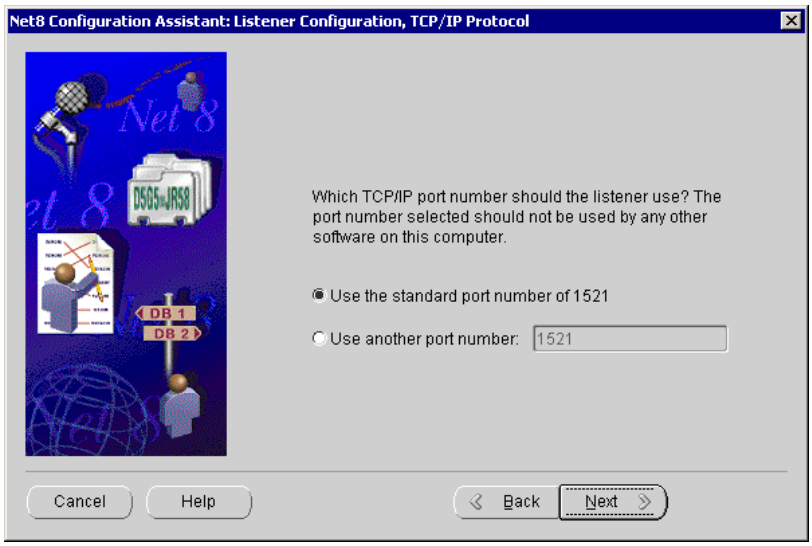


Figure 31. Port number selection

20. It is important that Oracle's listener service be dedicated to a unique port. At the TCP/IP protocol use the default port 1521 (Figure 31) unless you will be using applications that will also try to access this port. Click **Next**.

21. Select **No** when asked if you would like to configure another listener. Click **Next**.

22. Click **Next** to continue the configuration. You will see a window similar to Figure 32.

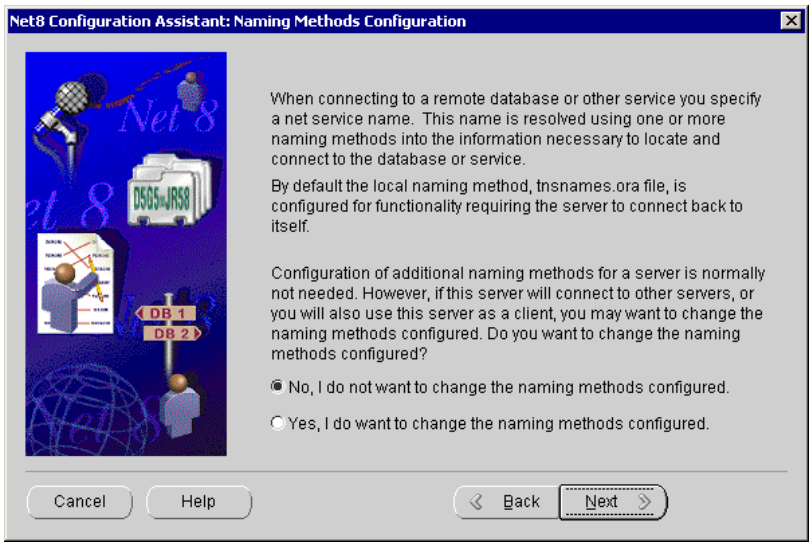


Figure 32. Naming Methods Configuration window

23. Select **No, I do not want to change the naming methods configured**. Click **Next**.

    Note: The naming method is to the Oracle databases what the Domain Name Server is to the TCP/IP. Configure it if you are in a network with other Oracle databases with naming methods configured.

24. On the Net8 Configuration Complete window, click **Finish**.

25. Wait until all tools statuses are successful before continuing. (See the example in Figure 33.)

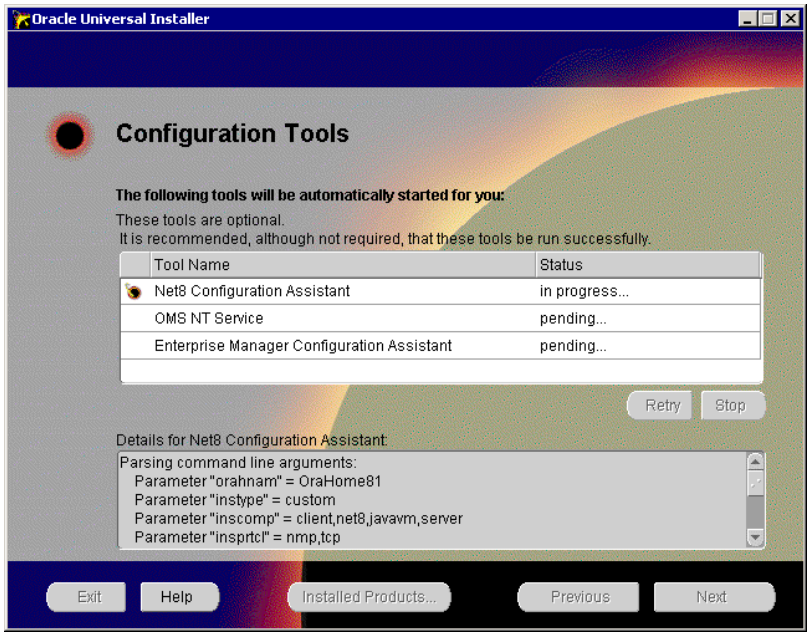


Figure 33. Configuration successful window

26. Check the log at the bottom of the window for the current status of the install. When all tools have succeeded, click **Exit**.

27. Edit the file SQLNET.ORA located in D:\Oracle\Ora81\Network\Admin (D:\Oracle is your Oracle directory), remarking the line

    ```
    SQLNET.AUTHENTICATION_SERVICES= (NTS)
    ```

    by placing the character # at the beginning of the line (see Figure 34)

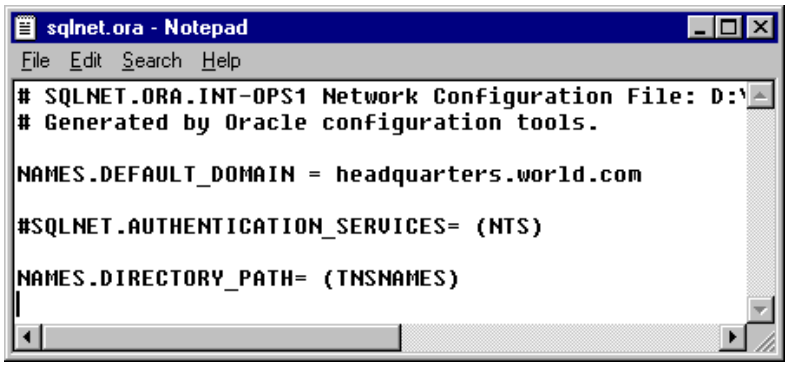


Figure 34. SQLNET.ORA window

28.Log off and log on all the nodes to load the new system variables.

7.3 Creating non-cluster aware services

Some of the services created by the Oracle8*i* installation are non-cluster aware. These services will need to be created individually on each node. One such service is the listener service. This service must be started for OPS to function correctly.

1. From node1, in our example, itsoops1, click **Control Panel -> Administrative Tools -> Component Management**, select **Services**. Check that the following service has started: *OracleOrahome81TNSListener*.

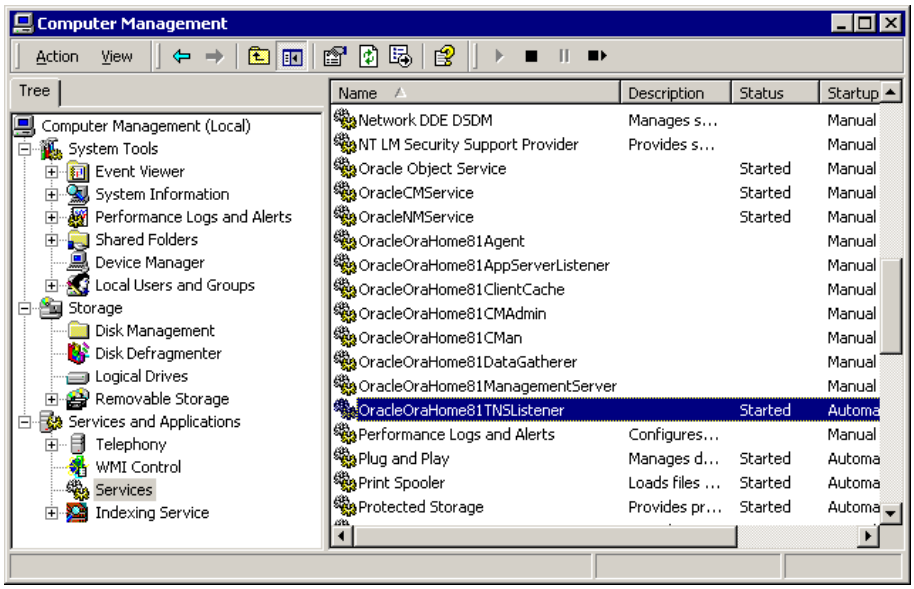


Figure 35. Oracle listener service on local node

2. Complete the following procedures on the second node.
   a. Open a command prompt, type `lsnrctl` and press Enter.
   b. After the listener control program starts, type `start listener` and press Enter. Trying to start the listener service will generate an error (1060). This is normal. This error triggers Oracle to create the listener service.

```
LSNRCTL> start LISTENER
Starting tnslsnr: please wait...

Failed to open service <OracleOraHome81TNSListener>, error 1060.
TNSLSNR for 32-bit Windows: Version 8.1.6.0.0 - Production
System parameter file is D:\Oracle\Ora81\network\admin\listener.ora
Log messages written to D:\Oracle\Ora81\network\log\listener.log
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\EXTPROC0ipc
)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=itsoops2.headquarters.wo
rld.com)(PORT=1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC0)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for 32-bit Windows: Version 8.1.6.0.0 - Produc
tion
Start Date                24-JUL-2000 23:54:13
Uptime                    0 days 0 hr. 0 min. 0 sec
Trace Level               off
Security                  OFF
SNMP                      OFF
Listener Parameter File   D:\Oracle\Ora81\network\admin\listener.ora
Listener Log File         D:\Oracle\Ora81\network\log\listener.log
Services Summary...
  PLSExtProc            has 1 service handler(s)
The command completed successfully
LSNRCTL> exit
```

Figure 36. Starting the listener service

c. After the command has completed, at the prompt type `Exit` and press Enter.

3. Open **Control Panel -> Administrative Tools -> Component Management**, select **Services**. Check that the following service has started: *OracleOrahome81TNSListener*.

Chapter 8. Creating and validating a database

This chapter explains which database application type to use and covers the creation of a simple database within the OPS environment. The activity in this chapter will validate your configuration. A database administrator should be readily available to render some assistance, if necessary.

8.1 Designing your database

The database design is the key to taking advantage of the scalability of the hardware and software of an Oracle Parallel Server system. Databases must be carefully designed to be scalable.

The main goal of the design is to minimize the computing resources used for parallel cache management by:

- Understanding the application functionality and access pattern
- Reducing the number of instance lock operations
- Taking advantage of cache fusion
- Partitioning the application and data

8.1.1 First considerations

Your priorities will decide the characteristics of your database. Questions to consider are:

- Is transaction speed more important than data safety?
- Will the database be used for a large number of short transactions or for a small number of large transactions?
- Is the database to be accessed by a single application or by multiple applications running on multiple nodes?

The system resources available to you will also affect your database installation, usage, and backup strategies. The number of users and applications running at one time will depend upon the physical memory available and the system running the database.

8.1.2 Database application type

The type of database application you are planning is a primary criterion in determining how you will plan and structure the database. Depending on the purpose for which they will be used, there are four basic types of database

applications: OLTP Database, Decision Support Database, Development Database, and Mixed Database.

8.1.2.1 OLTP database

An OLTP database, such as a bank automated teller machine (ATM), has a very high volume of transactions (measured in transactions per second). When planning a database for such purposes, you usually need to:

- Split database files so that disk I/O is shared among many disk subsystems.
- Implement redo log archiving.
- Be available 24 hours a day, 7 days a week.
- Use raw disk partitions.
- Use high availability hardware (mirrored disks, for example).
- Institute performance tuning.
- Set the INIT.ORA SGA parameters larger than the default values to accommodate more users and increase performance.

8.1.2.2 Decision support database

A decision support database, such as an inventory system, has a relatively low number of database updates (measured in transactions per hour). The users tend to make few queries, and they look at the results of these queries for many minutes at a time. They may prepare reports from these queries. A decision support database does the following:

- Splits database files so that disk I/O is shared among many disk subsystems
- Has less need to split the logical database design
- Has less need to implement redo log archiving
- Can usually be brought down for maintenance or backups
- Needs a larger SGA, because the volume of handled data is larger
- Table partitioning and materialized views will increase performance

8.1.2.3 Development database

A development database is used to develop new applications. Such a database does the following:

- Does not usually need to split I/O
- Can usually be brought down for maintenance or backups

- Has little need for backing up more than once a day
- Has little need for performance tuning

8.1.2.4 Mixed database

A mixed database combines various functions, such as decision support and transaction processing. Most databases fall within this category. Such a database:

- Has an even mixture of queries and updates
- Has some performance tuning needs, but not as much as an OLTP database

8.1.3 Tablespaces design

Create separate tablespaces to hold the following categories of segments. For each categories, separate the indexes from the data into different tablespaces:

- Read-only data
- Data that is modified by only one instance
- Data that is modified by only one instance but read by others
- Data that is modified by many instances
- Rollback segments
- Temporary segments
- Miscellaneous

This will allow you to better administer the PCM locks on the data.

For more information on how to administer the PCM locks, see Chapter 7, *Planning the Use of PCM and Non-PCM Instance Locks* of the PDF Oracle8*i* Parallel Server Administration, Deployment, and Performance. This is located on the Oracle8*i* Server online documentation CD under Oracle8*i* Parallel Server, Release 8.1.6.

Each instance will have its own set of redo logs.

8.1.4 CM disk

OPS can be configured to use a Cluster Manager disk for the integrity of the data. To determine whether a node within a cluster is alive we use one component of the OSD, the Cluster Manager (CM) component. For communication with the other nodes CM uses the dedicated network

connectivity (actually a 10/100 Ethernet LAN). The loss of network connectivity can result in two or more separate subclusters operating independently upon the same shared database. This condition, commonly known as cluster partitioning or a split-brain cluster, must be prevented by the OSD-CM. If a cluster becomes partitioned and is allowed to continue operations, the operation of the nodes is no longer synchronized, resulting in corrupted data. This problem is addressed by using the shared disk to determine which nodes are alive, independent of network connectivity. Each node that is a member of the cluster will write a heartbeat to the shared disk. All nodes will be able to observe these heartbeats and determine if another node is alive. At critical points when a node is either joining or leaving a cluster, a partition check will be performed to see if the current node has become partitioned and should shut itself down.

For the CM disk, plan an 8 MB partition on the shared disks.

8.1.5 Planning your database

In this section, we will design a database for basic usage. It is the database we have created in our labs. We have named it ITSO. Adapt it to your specific requirements.

Develop an outline of your database files (data, control and redo log files) to include their file size.

The symbolic names are the name of the raw partitions (where you are going to store the database files) that Oracle will understand. You have to choose values for these symbolic names; we are going to assign these names to the respective raw partition later in 8.4, “Create symbolic links” on page 107. For the naming purposes, we recommend that you put the name of the database at the beginning of the symbolic name, and choose a name that reflect what the partition contains. For example, for the System tablespace, we have chosen ITSO_system1.

The Cluster Management disk symbolic name is the value chosen during the OSD installation; see 6.1.2, “Installation on the first node” on page 59. For our example, the Cluster Management disk symbolic name is ITSO_CMdisk.

Create a table containing the information on these files, the associated tablespaces, their Oracle symbolic name, their size, and the description of their content (see Table 2).

Note: You can set your redo log files to a lower value than the one indicated in Table 2, but the minimum size of a partition you can create with Windows 2000 is 8 MB.

Table 2. Description of the database files

Tablespace	Symbolic name	Size	Description
Data Files			
SYSTEM	ITSO_system1	150 MB	System data
TOOLS	ITSO_tools1	150 MB	Miscellaneous
USERS	ITSO_users1	100 MB	User workspace
INDEXES	ITSO_indx1	50 MB	Indexes
TEMP	ITSO_temp1	100 MB	Temp data
RBS	ITSO_rbs1	50 MB	Rollback data
Control Files			
	ITSO_control1	105 MB	First control file
	ITSO_control2	105 MB	Second control file
Redo Log Files			
	ITSO_redo1_1	10 MB	1st log file for ITSOOPS1 thread 1 member 1
	ITSO_redo1_2	10 MB	2nd log file for ITSOOPS1 thread 1 member 1
	ITSO_redo2_1	10 MB	1st log file for ITSOOPS2 thread 2 member 1
	ITSO_redo2_2	10 MB	2nd log file for ITSOOPS2 thread 2 member 1
	ITSO_redo1_1bis	10 MB	1st log file for ITSOOPS1 thread 1 member 2
	ITSO_redo1_2bis	10 MB	2nd log file for ITSOOPS2 thread 1 member 2
	ITSO_redo2_1bis	10 MB	1st log file for ITSOOPS2 thread 2 member 2
	ITSO_redo2_2bis	10 MB	2nd log file for ITSOOPS2 thread 2 member 2
Cluster management			
	ITSO_CMdisk	8 MB	CM disk

Note: Windows 2000 didn't allow us to create a CM disk size less than 8 MB.

8.2 Create RAID arrays

In the following sections, we will review several recommendations about the different RAID levels, and then discuss how to create and modify logical drives.

8.2.1 Which RAID level to choose?

With the Fibre RAID controller it's possible to connect up to six Netfinity EXP200 disk enclosures, for a total of 60 hard disk drives. For data redundancy, we can use different RAID levels. The Fibre Channel RAID Controller allows you to use four different RAID levels:

- RAID-0: With Raid-0 the data is striped across multiple disk drives without parity protection. This means that if you have a drive failure on one hard disk, it will result in a complete failure of the array and all data is lost. However, RAID-0 is the fastest RAID level available. RAID-0 also has the lowest cost of implementation.
- RAID-1: RAID-1 provides fault tolerance by employing a mirror drive for every data drive. The mirror drive ensures access to data should any single drive fail. RAID-1 is the most expensive way to implement RAID. RAID-10 is RAID-1 on more than two disks, the data is striped and mirrored. Netfinity Fiber Channel Storage Manager understands RAID-10 when you create a four-disk RAID-1 array. RAID-1 and RAID-10's performance is not as good as RAID-0 but better than both RAID-3 and RAID-5.
- RAID-3: RAID-3 is the same as RAID-5, but the checksum is only on one drive. RAID-3 is better suited for larger record performance. RAID-5 is better suited for small records, such as those used in transaction processing.
- RAID-5: In a RAID-5 configuration the data parity is striped over all the drives in the LUN. Because of the parity, you can afford a single drive failure. Data can be still be recovered from the remaining good drives. Two drive failures cause all data to be lost. A significant benefit of RAID-5 is the low cost of implementation, especially for configurations requiring a large number of disk drives.

For more specifications, see Table 3 and Table 4:

Table 3. RAID performance characteristics

RAID Level	Data Capacity	Large Transfers		Small Transfers		Data Availability
		Read	Write	Read	Write	
Single Disk	n	Good	Good	Good	Good	Fair
RAID-0 RAID-10	n	Very Good	Very Good	Very Good	Very Good	Poor
RAID-1	n/2	Very Good	Good	Very Good	Very Good	Very Good
RAID-3	n-1	Good	Fair	Good	Fair	Very Good
RAID-5	n-1	Good	Fair	Good	Fair	Very Good

Table 4. Disk subsystem rules of thumb

Performance of this configuration	Is equivalent to...
RAID-0	50% more throughput than RAID1 (same number of drives)
RAID-1	50% more throughput than RAID5 (same number of drives)
Doubling the number of drives	50% increase in drive throughput (until disk controller becomes a bottleneck)
One 10.000 (10K) RPM drive	10-50% improvement over 7200 RPM drives (50% when considering RPM only, 10% when comparing with 7200 RPM drives with rotational positioning optimization)

8.2.2 Recommendations

Here are several recommendations for configuring a Fibre Channel and shared storage solution.

8.2.2.1 An array

An array is a physical set of drives in the RAID controller. An array is composed of several hard disk drives. In an array, we can create several logical partitions but the RAID level must be the same for all logical partitions. With Windows 2000 we cannot have more than eight configured drive groups.

We also recommend that you:

- Distribute the ownership of the arrays to the different controller units to avoid any bottleneck (if you have redundant controllers). For example, CMdisk is assigned to slot A and Data is assigned to slot B.
- Separate the data files, redo log files, and index files on different arrays.

- Stripe the data files, index files, rollback segment files, and temporary files in multiple-disk arrays.

8.2.2.2 Number of logical partitions

Windows 2000 limits the maximum number of logical partitions per RAID controller unit to eight, whether the module has a single or redundant controllers. If you need to create logical partitions on arrays of more than 20 drives (up to 30), you must do so from the command line, using the RAIDUTIL utility.

8.2.2.3 Description of our array/RAID configuration

In our configuration, we used RAID-0 and RAID-1. We chose these RAID levels because they are fault tolerant and cost effective. However, it depends on your configuration and your database. If you want to have maximum performance without fault tolerance, the best RAID level is RAID-0. On the other hand, if you want fault tolerance but high performance, RAID-1 is a better choice. RAID-0 is for our redo logs and RAID-1 is for CMdisk, data and index.

8.2.2.4 Hot spare

When you use RAID 1, 3 or 5 it is important to have a hot spare drive in your configuration. When using RAID-3 and RAID-5, if more than one disk fails you will lose all the LUNs created on these groups or arrays. We therefore recommend that you configure at least one hot spare drive per EXP unit. In case of a drive failure, the data from the failed drive will automatically be rebuilt to the hot spare drive. The time it takes to rebuild depends on the size of the drives.

8.2.2.5 Multiple EXP configuration

If you have more than one EXP in your configuration, we recommend that you create the logical drives based on disks located in different EXP200s. For example, we can create a RAID-1 LUN with four disks located in two different EXP200. The data drive and the mirrored drive are not in the same enclosure. So in case of an EXP200 failure we can still access to the data.

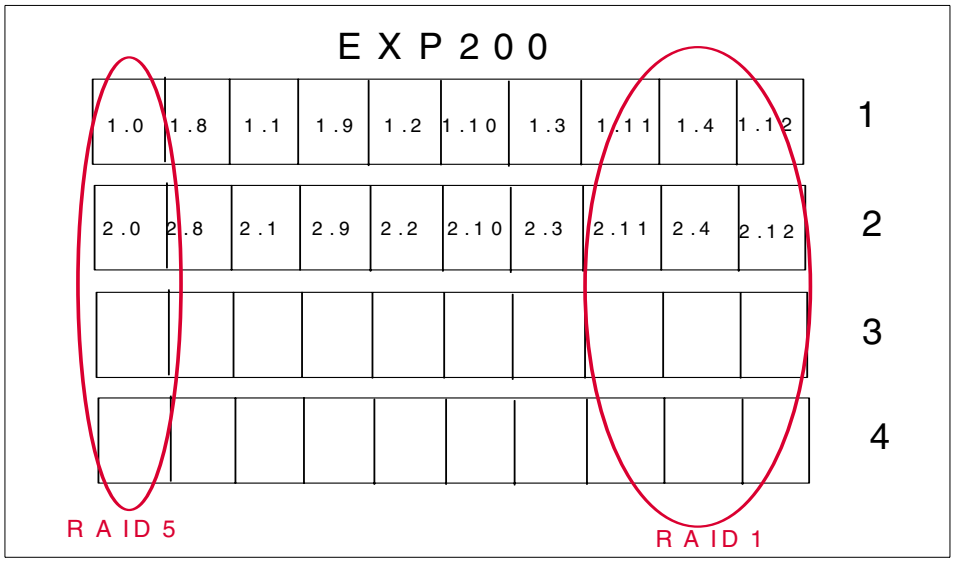


Figure 37. Different RAID configurations

The drives are assigned a unique ID that is composed of two numbers. The first one is the channel on which the EXP200 is connected (up to six) to the RAID controller unit, and the second one is the ID of the drive in the EXP200. For maximum fault tolerance and performance it is possible to create an array (group) composed of 12 drives in four different EXP200s in a RAID-1 configuration. For example, drive 1.11 will be mirrored on the drive 2.11, and 3.11 will be mirrored on 4.11. So in that case, if up to two EXP200s fail, we can still access the RAID-1 data (see Figure 37). In a RAID-5 configuration we can have the same fault tolerance but only if we have no more than one drive in each EXP200 in the array.

8.2.2.6 Segment size

Six different segment sizes exist: 8,16, 32, 64, 128 and 256 KB. For OPS, we recommend that you use the size of 16 KB.

8.2.2.7 Caching parameters

Each RAID module has 128 MB of cache memory. By using caching you can increase system performance because the data for a read from the host may already be in the cache from a previous operation (thus eliminating the need to access the drive itself), and a write operation is completed once it is written to the cache, rather than to the drives.

Three cache options are available with the storage management software:

- Write caching: This option enables the use of the cache that allows increased performance.
- Write cache mirroring: This option allows cached data to be mirrored across two redundant controllers with the same cache size.
- Cache without batteries: Allows write caching to continue even if the batteries are discharged completely or not fully charged, or if there are no batteries present. Use this option only if you have a UPS for power backup.

We recommend that you use the write caching option for your data and indexes arrays. Do not use the Write caching for the log arrays, unless you have a UPS. In this case, use cache without batteries. That will allow you to increase the performance while protecting your data. If the power fails, the logs will be written on the disk, so you will not lose any data. Of course, if you have a UPS, you will be able to use the cache without batteries for all the arrays, so you will have the best performance.

8.2.3 Planning your arrays

Create a table containing the information from Table 2 on page 81, and associate to a RAID Array/Logical Volume and RAID level to the database files. See Table 5 on page 87.

For our database, we chose to put the data on a RAID-1 array, the indexes on another RAID-1 array, the CM disk on a RAID-1 array, and the log files on two RAID-0 arrays. The two members of each redo log groups reside on different arrays (Oracle redo log mirroring).

Table 5. Description of the database files and their arrays

Array/ Logical Volume	Raid level	Tablespace	Symbolic name	Size	Description
Data Files					
data	1	SYSTEM	ITSO_system1	150 MB	System data
data	1	TOOLS	ITSO_tools1	150 MB	Miscellaneous
data	1	USERS	ITSO_users1	100 MB	User workspace
indexes	1	INDEXES	ITSO_indx1	50 MB	Indexes
data	1	TEMP	ITSO_temp1	100 MB	Temp data
data	1	RBS	ITSO_rbs1	50 MB	Rollback data
Control Files					
data	1		ITSO_control1	105 MB	First control file
indexes	1		ITSO_control2	105 MB	Second control file
Redo Log Files					
log1	0		ITSO_redo1_1	10 MB	1st log file for ITSOOPS1 thread 1 member 1
log1	0		ITSO_redo1_2	10 MB	2nd log file for ITSOOPS1 thread 1 member 1
log1	0		ITSO_redo2_1	10 MB	1st log file for ITSOOPS2 thread 2 member 1
log1	0		ITSO_redo2_2	10 MB	2nd log file for ITSOOPS2 thread 2 member 1
log2	0		ITSO_redo1_1bis	10 MB	1st log file for ITSOOPS1 thread 1 member 2
log2	0		ITSO_redo1_2bis	10 MB	2nd log file for ITSOOPS2 thread 1 member 2
log2	0		ITSO_redo2_1bis	10 MB	1st log file for ITSOOPS2 thread 2 member 2
log2	0		ITSO_redo2_2bis	10 MB	2nd log file for ITSOOPS2 thread 2 member 2
Cluster Management					
CM Disk	1		ITSO_CMdisk	8MB	CM disk

8.2.4 Creation of the logical drives and hot spare drive

This section describes how to create logical drives and a hot spare drive.

Using Table 5, complete the following steps:

In our example, we installed Netfinity Fibre Channel Storage Manager client on our second node, itsoops2.

1. Start the Netfinity Fiber Channel Storage Manager 7 client on the machine dedicated for storage management (in our example, itsoops2). You will see a window similar to Figure 38.

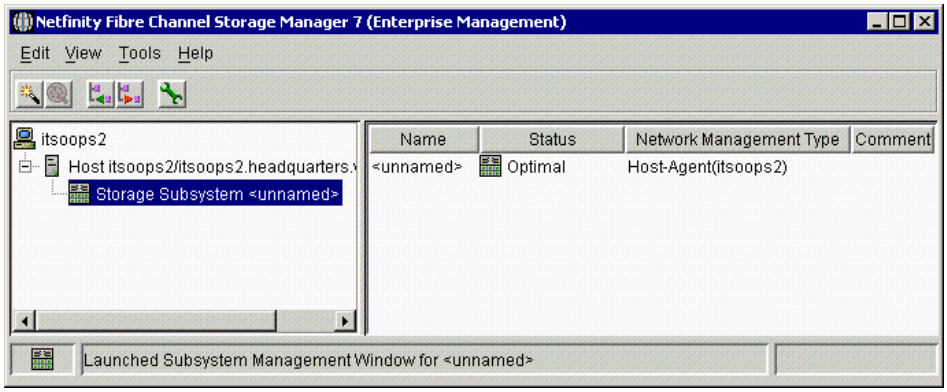


Figure 38. Netfinity Fibre Channel Storage Manager 7 window

2. Double-click **Storage Subsystem**. You will see a window similar to Figure 39.

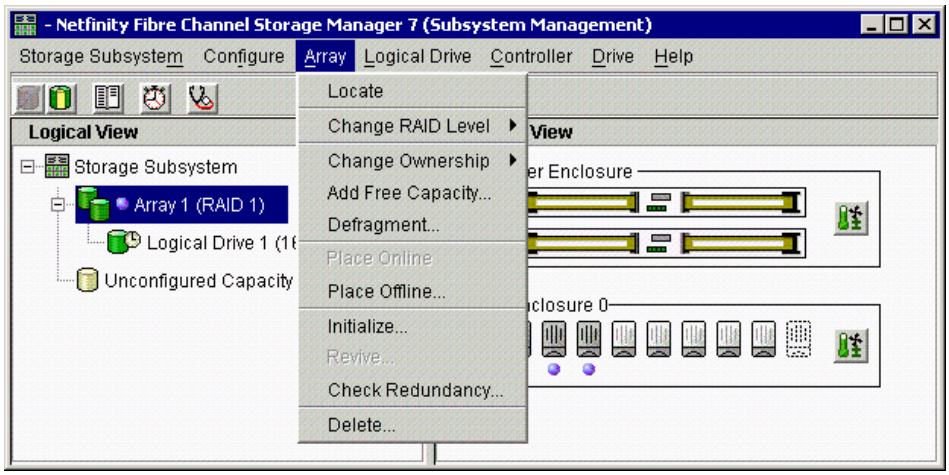


Figure 39. Storage Subsystem's Array Menu window

3. Remove any existing arrays. To do so, select the existing array(s) and from the Array menu, click **Delete**. Click **Yes** to confirm. You will see a window similar to Figure 40.

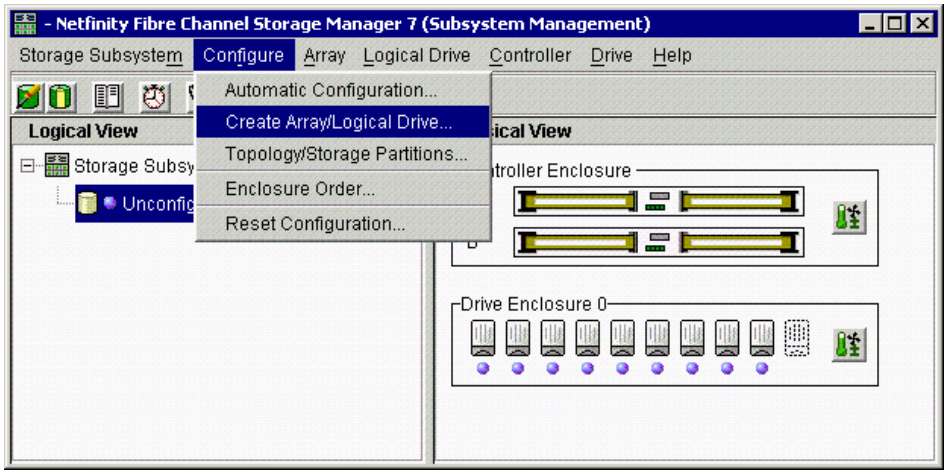


Figure 40. Storage Subsystem's configure drop-down

4. Now, all the storage is available for use. Create the logical drives for the CM_disk, Data, and Indexes. From the Logical View, highlight **Unconfigured Capacity** and begin creating the CM_disk logical drive.
5. From the action bar, click **Configure -> Create Array/Logical Drive**. You will see a window similar to Figure 41.

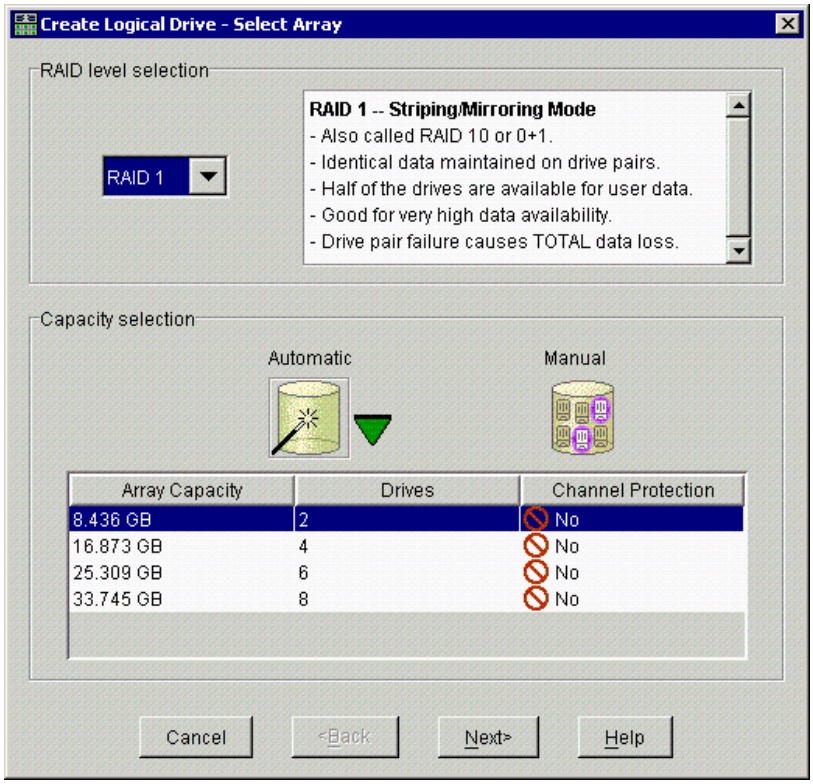


Figure 41. Create Logical Drive - Select Array window

6. Select the RAID level and the number of disks the array will contain. For our CM_disk, the RAID level is 1, utilizing two drives.

   Once the array is created, it is not possible to decrease the number of drives assigned to the array, but it is possible to increase it.

   Note: Select **Manual** only to specify which disk will be part of the array.

7. Click **Next** to continue. You will see a window similar to Figure 42.

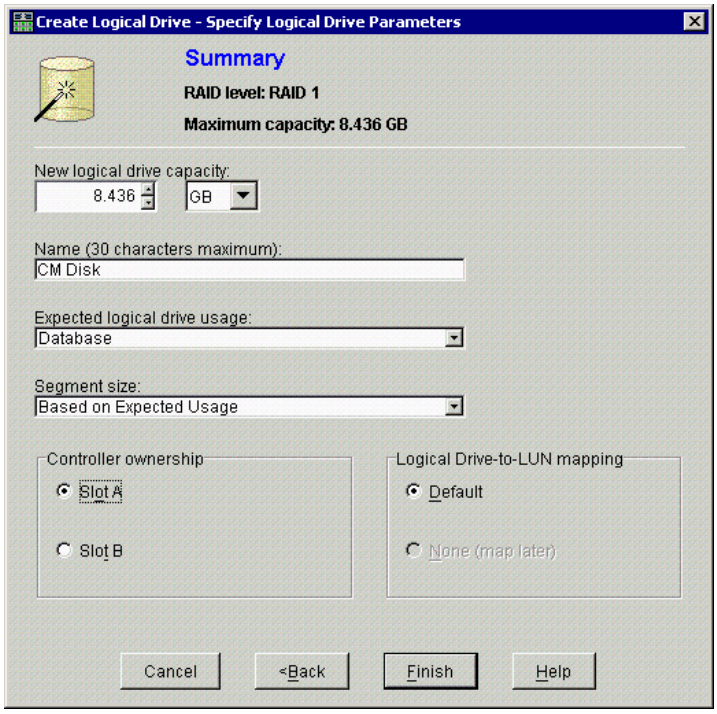


Figure 42. Specify Logical Drive Parameters window

8. Verify that the maximum drive capacity is listed in the New logical drive capacity field.
   Note: You can have more than one logical drive in the array. If you set your logical drive capacity to less than the maximum of the array, you will be able to create other logical drives in the array. It will not be possible to change the logical drive capacity after the creation.
9. Type the name of your logical drive in the Name field. For our example, type `CM_Disk`.
10. In the Expected logical drive usage field, select **Database**.
11. In the Segment size field, select **Based on Expected Usage** (size defaults to 16 KB when Database is selected).
12. Select which controller slot will have the ownership of the array.

    Note: To get better performance, do not assign all logical partitions to the same RAID controller unit. In our example, we selected **Slot A**.
13. Click **Finish** and click **OK** to confirm. The new array will appear in your Storage Subsystem window.

14. Right-click the new array and select **Properties**. You will see a window similar to Figure 43.

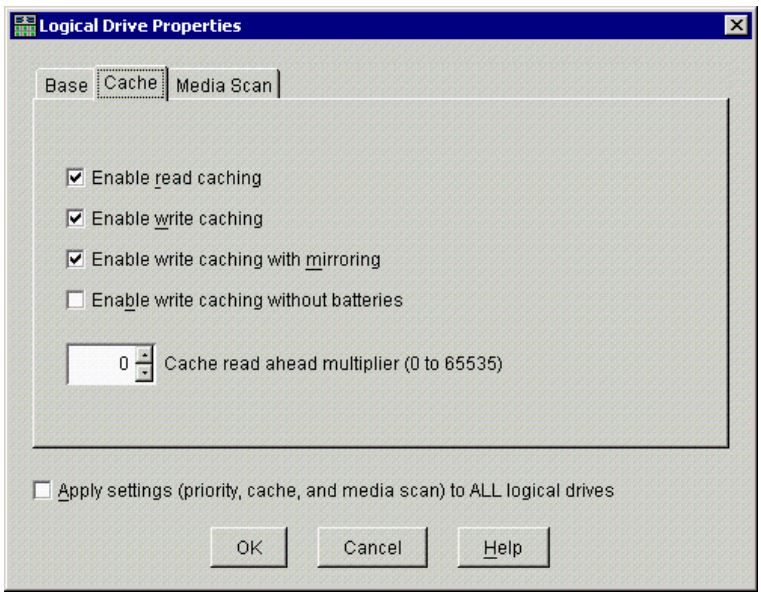


Figure 43. Logical Drive Properties window

15. Select the **Cache** tab and make sure **Enable write caching** is checked.
16. Click **OK**.
17. Repeat step 4 on page 89 through step 16 on page 92 to create the Data and Indexes logical drives.
18. After the Data and Indexes logical drives have been created, create your logical drives for the logs. In the Logical View window, highlight **Unconfigured Capacity**.
19. From the action bar, click **Configure -> Create Array/Logical Drive**. You will see a window similar to Figure 44.

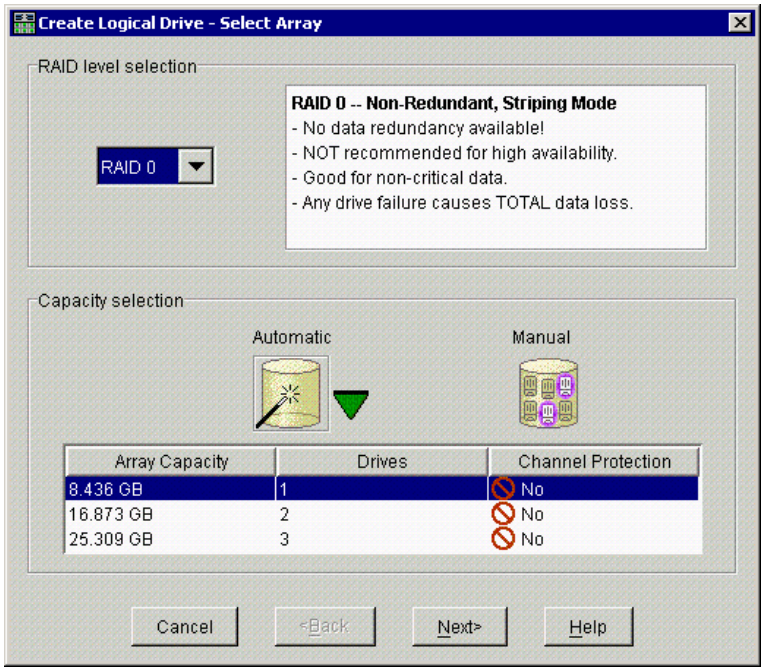


Figure 44. Create Logical Drive Select Array window

20. Select the RAID level and the number of disks the array will contain. In our example, the RAID level is 0, and the number 1 drive is selected.
21. Click **Next**. You will see a window similar to Figure 44.

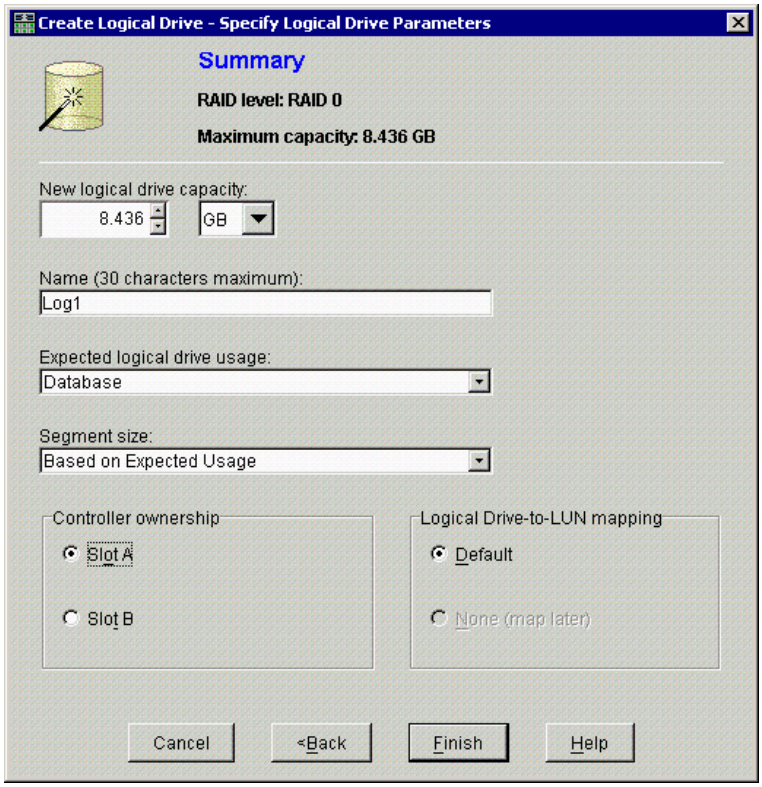


Figure 45. Specify Logical Drive Parameters window

22. Verify the maximum drive capacity is listed in the New logical drive capacity field.
23. Type the name of your logical drive in the Name field. For our example, type `Log`.
24. In the Expected logical drive usage field, select **Database**.
25. In the Segment size field, select **Based on Expected Usage** (size defaults to 16 KB when Database is selected).
26. Select which controller slot will have the ownership of the array to get better performance, do not assign all logical partitions to the same RAID controller unit. In our example, we selected **Slot A**.
27. Click **Finish** and click **OK** to confirm. The new array will appear in your Storage Subsystem window.
28. Right-click the new array and select **Properties**. You will see a window similar to Figure 46.

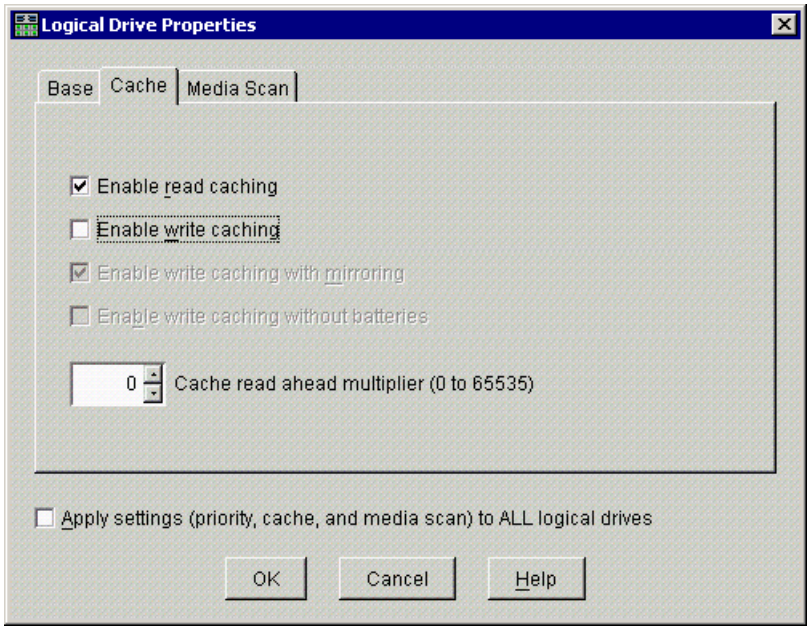


Figure 46. Logical Drive Properties window

29. Click the **Cache** tab. Uncheck **Enable write caching** and click **OK**.
30. Repeat step 18 on page 92 through step 29 on page 95 once more to create a second logical drive for logs.
31. Create one hot spare drive per EXP200. For maximum fault tolerance, choose drives from different EXP200s.
32. Return to the Subsystem Management window (see Figure 47).

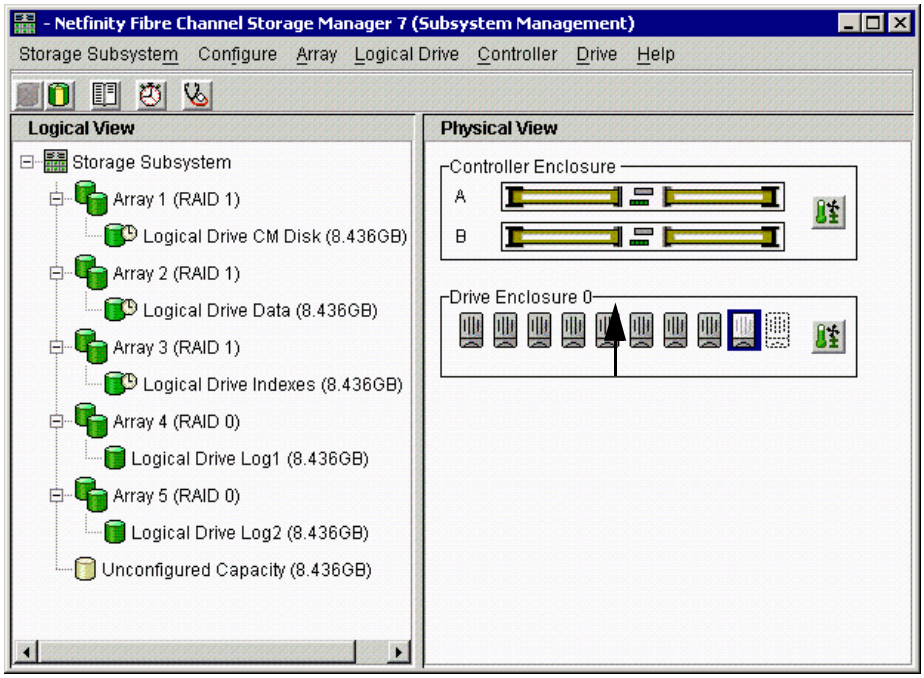


Figure 47. Subsystem Management window

33. In the Drive Enclosure pane in the Physical View, select a drive that has not been assigned to an array (white and grey colored).
34. From the action bar, click **Drive -> Assign Hot Spare**.

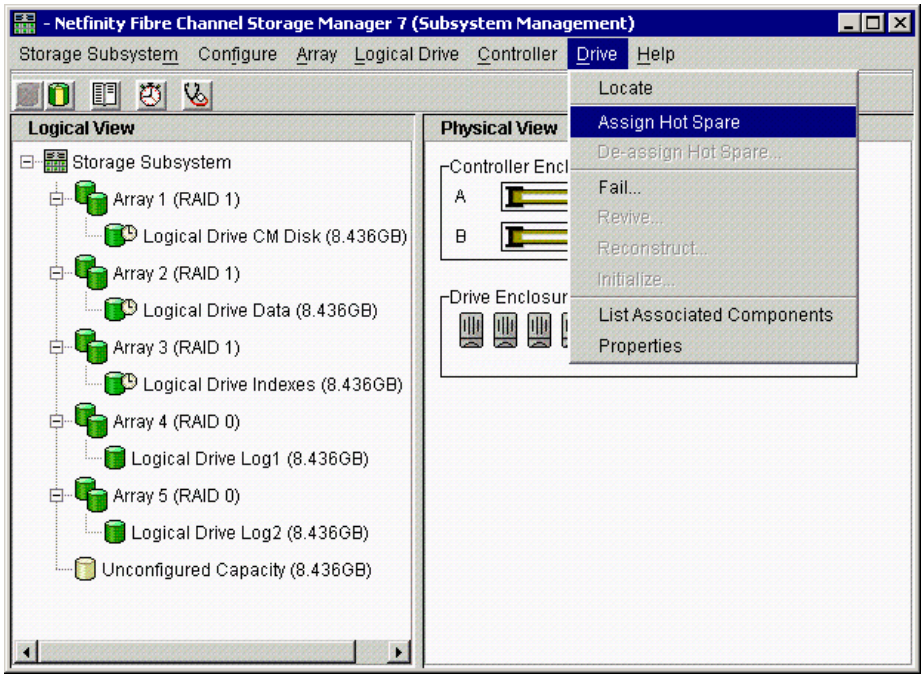


Figure 48. Subsystem Management window

35.Close the Fiber Channel Storage Manager, and reboot the two nodes.

8.2.5 Modification of logical drives

When you have created an array/logical drive, you can modify the following:

- Change the RAID level: You can perform this operation dynamically, but you are limited by the number of drives.
- Add drives: You can add drives in a group. But it's impossible to increase the logical drive size.
- Modify segment size: You can modify the segment size of individual logical units in a group. The available sizes are: 8, 16, 32, 64, 128 and 256. You can choose only the first upper and the first lower segment size of the existing size. For example, if the current size is 64 you can choose only 32 or 128. So if you want a segment size of 16 you must do that in two steps.
- Defragment the logical drives.
- Change the cache parameters: Use the Maintenance and Tuning application.

- Change the reconstruction rate: Use the Maintenance and Tuning application.

8.3 Implementing raw partitions

We have set up the external disk subsystem using Netfinity Fibre Channel Storage Manager (see Figure 47 on page 96). We created five logical drive partitions that are seen by Windows 2000 as five different drives. Now we are going to create the raw partitions (see 2.4.2, “Raw partitions” on page 23) we are going to store the database files.

Using Table 5 on page 87, follow these steps to create the raw partitions using Windows 2000:

1. On the first node only, start the Computer Management tool. Select **Disk Management**. You will see a window similar to Figure 49.

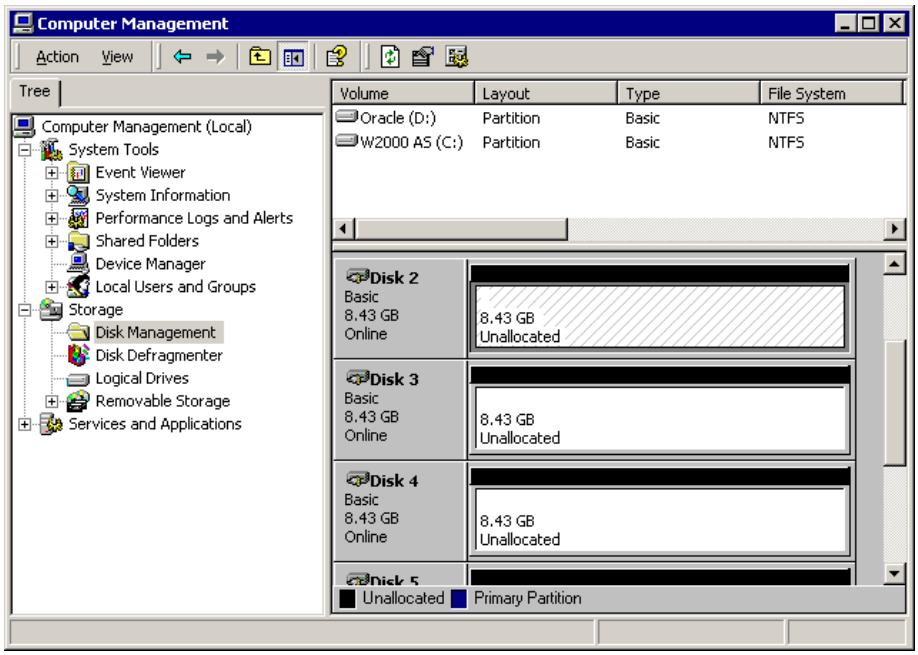


Figure 49. Disk Management window

2. The Write Signature and Upgrade Disk Wizard will appear unless it was turned off. Right-click on a disk that has not been signed (with a one-way roadsign). Select **Write Signature**. Select all the drives that have not been signed. Click **OK**.

Note: The disks are presented in the same order as in the Fiber Channel Storage Manager.

3. In the next step, we will create extended partitions on each logical drive (for example, Disk 2, Disk 3, Disk 4, etc.). From the Disk Management window, right-click Disk 2. You will see a window similar to Figure 50.

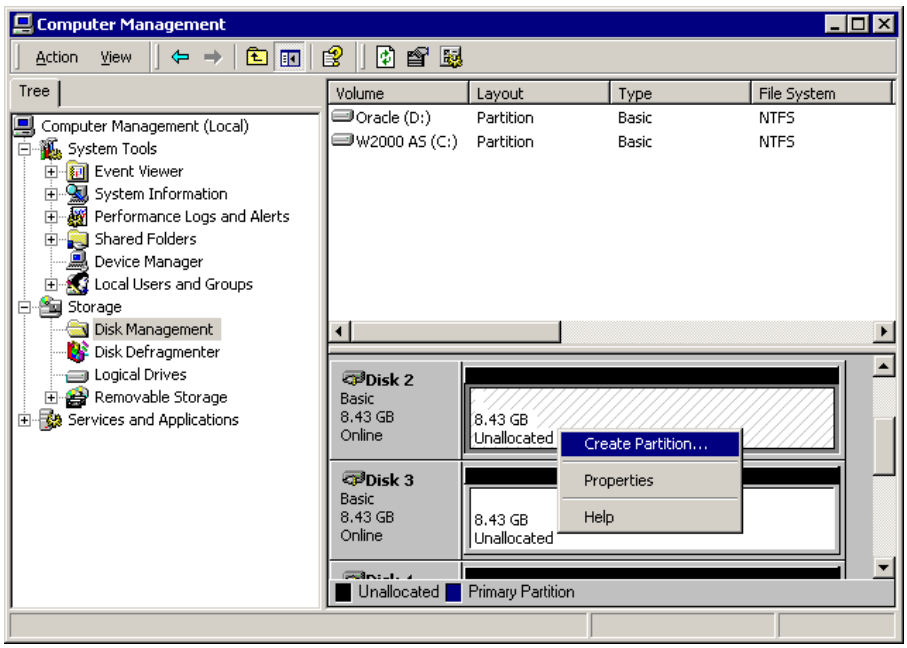


Figure 50. Create Partition window

4. Select **Create Partition** and click **Next**. Click **Next** in the Wizard. You will see a window similar to Figure 51.

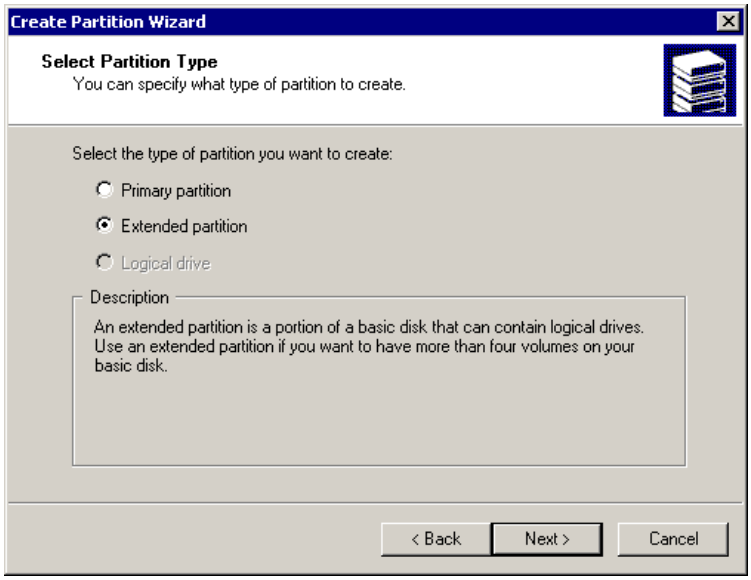


Figure 51. Extended partition window

5. Select **Extended partition**, then click **Next**. You will see a window similar to Figure 52.

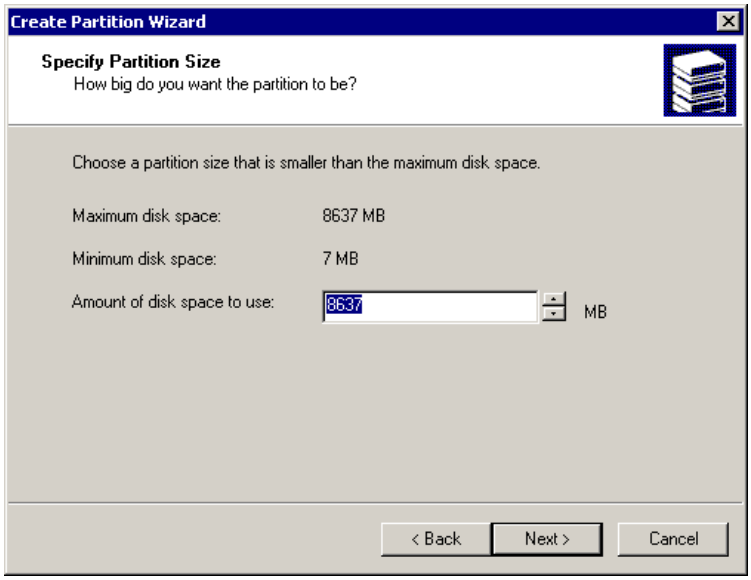


Figure 52. Create Partition Wizard window

6. Enter the amount of space the extended partition will contain. You should specify the entire disk space available, for example, 8637 MB. Click **Next**.
7. Click **Finish**.
8. Repeat step 3 on page 99 through step 7 on page 101 for the remaining (Fibre Channel) logical drives.

   In the next step, we will create the (Windows) logical drives. The number of logical drives required is dependent on the number of OPS nodes within the cluster, because each instance has its own redo log groups. For our example, we have two nodes.
9. From the Disk Management window, right-click an extended partition, for example, Disk 2. You will see a window similar to Figure 53.

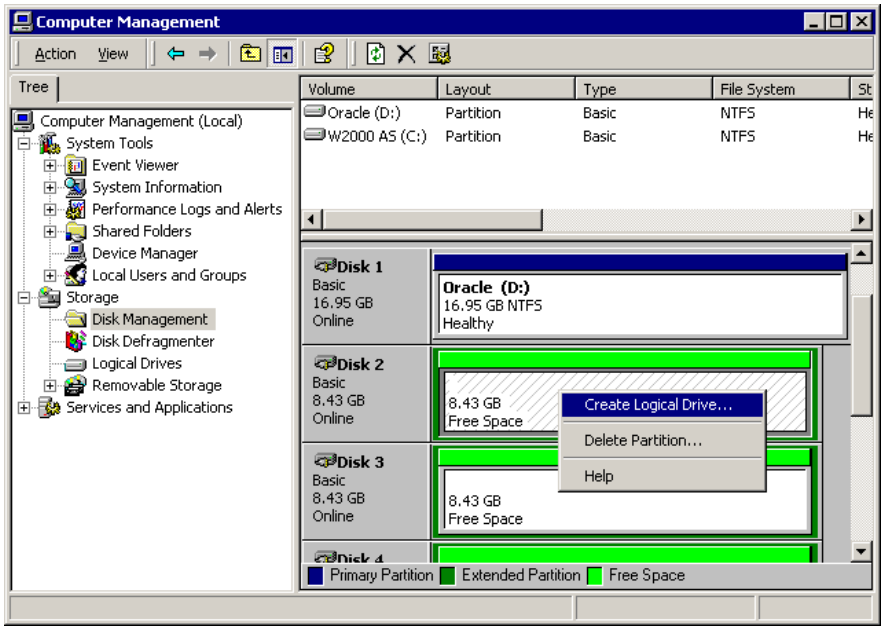


Figure 53. Create Logical Drive window

10. Click **Create Logical Drive** and click **Next**. You will see a window similar to Figure 54.

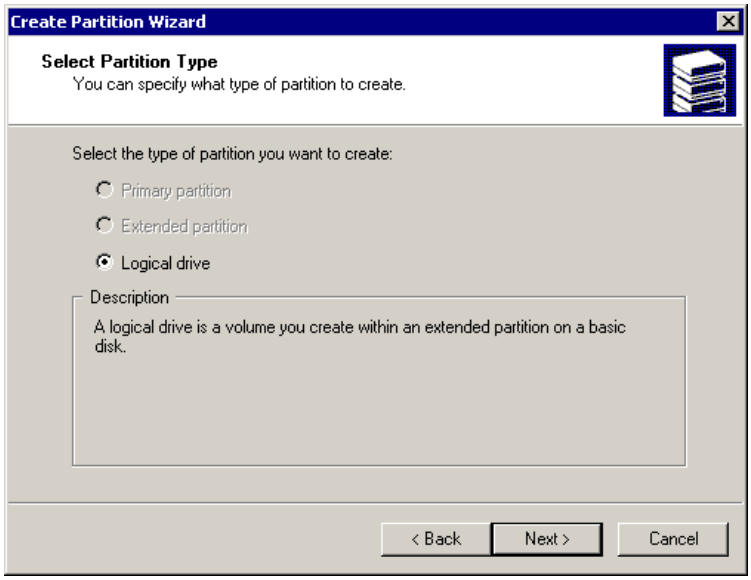


Figure 54. Create Partition Wizard window

11. Select **Logical Drive** and click **Next**. You will see a window similar to Figure 55.

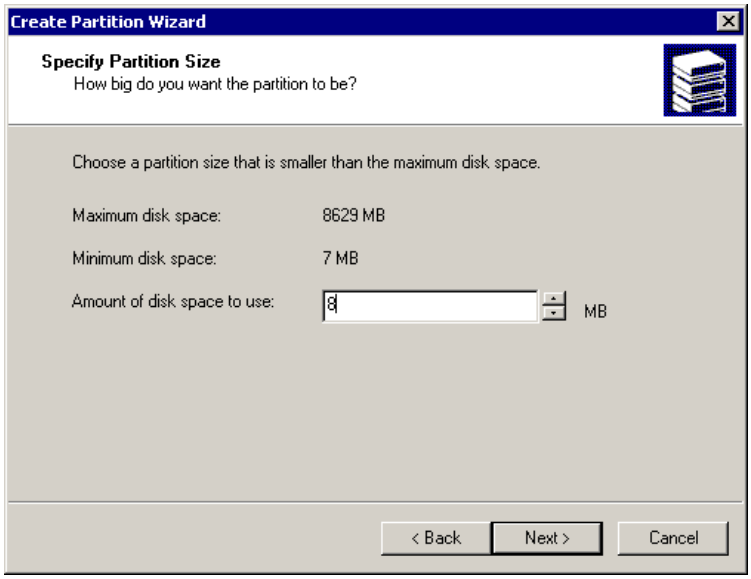


Figure 55. Specify Partition Size window

12. Enter the size of the partition. This should be the size of the tablespace that will be stored in the raw partitions: an additional 2 or 3 MB. For our example, we entered 8. Click **Next**. You will see a window similar to Figure 56 on page 104.

    Note: Windows 2000 will create the partition with two or three megabytes less or more than the size you have specified. Make sure the size will be enough for your tablespace requirement. If not, delete it and recreate it with a slightly larger size.

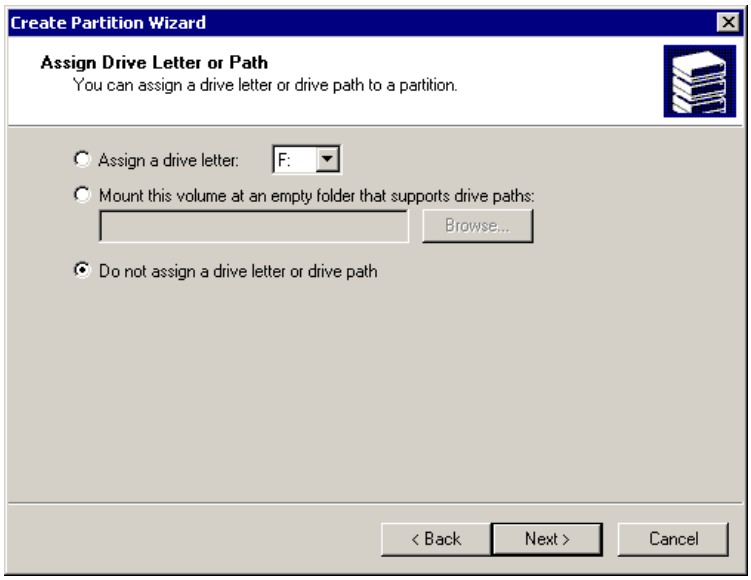


Figure 56. Assign Drive Letter or Path window

13. Select **Do not assign a drive letter or drive path**. Click **Next**. You will see a window similar to Figure 14.

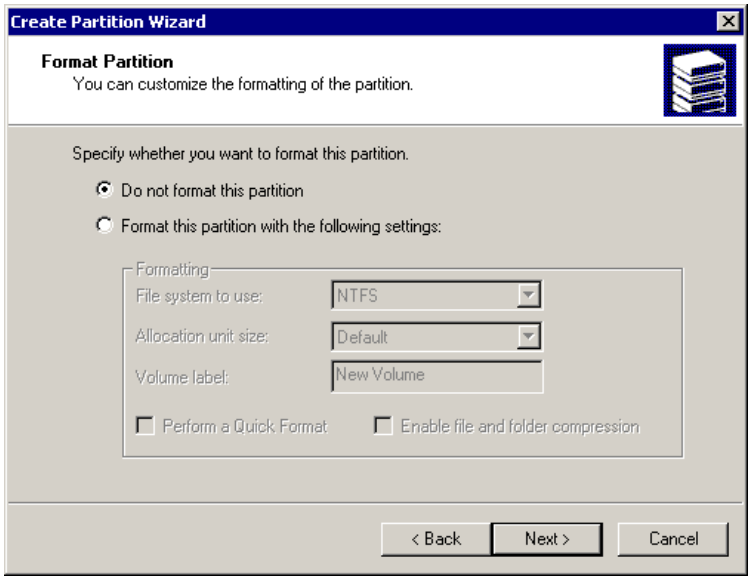


Figure 57. Format Partition window

14. Select **Do not format this partition** and click **Next**.
15. Click **Finish**.
16. Repeat steps 9 on page 101 through 15 on page 105 for all your extended partitions.
17. This completes the steps for the first node. Close the Computer Management tool.
18. On the second node, start the Computer Management tool. Select **Disk Management** (see Figure 58).
19. The Write Signature and Upgrade Disk Wizard will appear unless you have it turned off. Right-click a disk that has not been signed (with a one-way road sign), and select **Write Signature**. Select all the drives that have not been signed. Click **OK**.

    Note: The logical partitions may not have the same hard disk number on the second node (each node), but they map to the same space on the shared disk.

    Oracle does not use drive letters to access the partition so we will unassign the drive letters of the logical partitions.

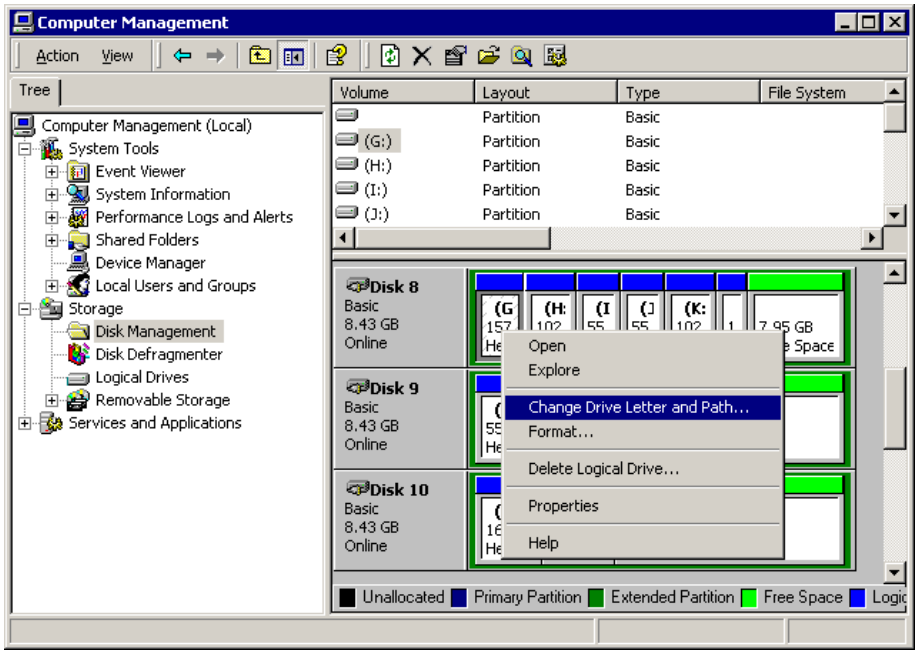


Figure 58. Change Drive Letter and Path window

20. Select a partition and right-click on the partition, select **Change Drive Letter and Path**. You will see a window similar to Figure 59.

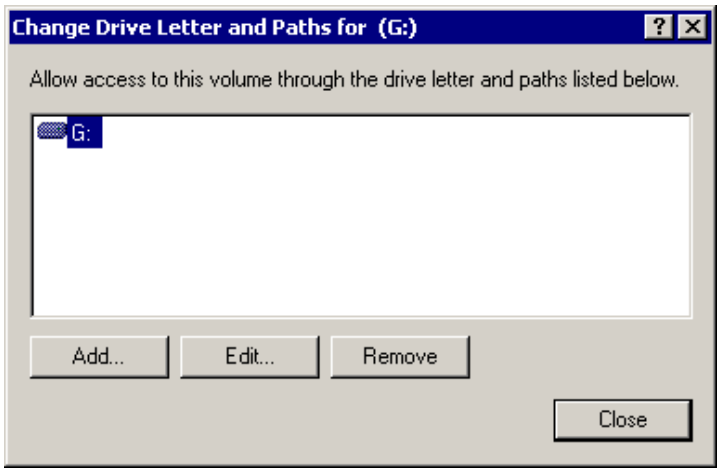


Figure 59. Change Drive Letter and Paths window

21. Repeat step 20 on page 106 for all the partitions.

    Note: If additional nodes were included in your configuration, you would complete these steps on each remaining node in the cluster.

8.4 Create symbolic links

Now we need to create links between the Oracle database and logical partitions. We will map the logical partitions to symbolic names. This has to be done on all the nodes. Using Table 5 on page 87, follow this process:

1. Create a text file and name it, for example, ITSO.tbl. We suggest you keep this file in an Admin directory under your Oracle directory (for example, D:\Oracle\Admin). Fill in this file with the symbolic name of the raw logical partitions that Oracle will understand and the operating system name of the raw logical partitions that Windows 2000 understands.

   Follow this model:
   <DB name>_<Raw Device name> \Device\Harddisk<N>\Partition<X>

   - N is the hard disk number (find this number in Disk Management), see Figure 60 on page 108.
   - X is the logical drive number (the first logical drive of the hard disk is number 1, the second is number 2, etc.)

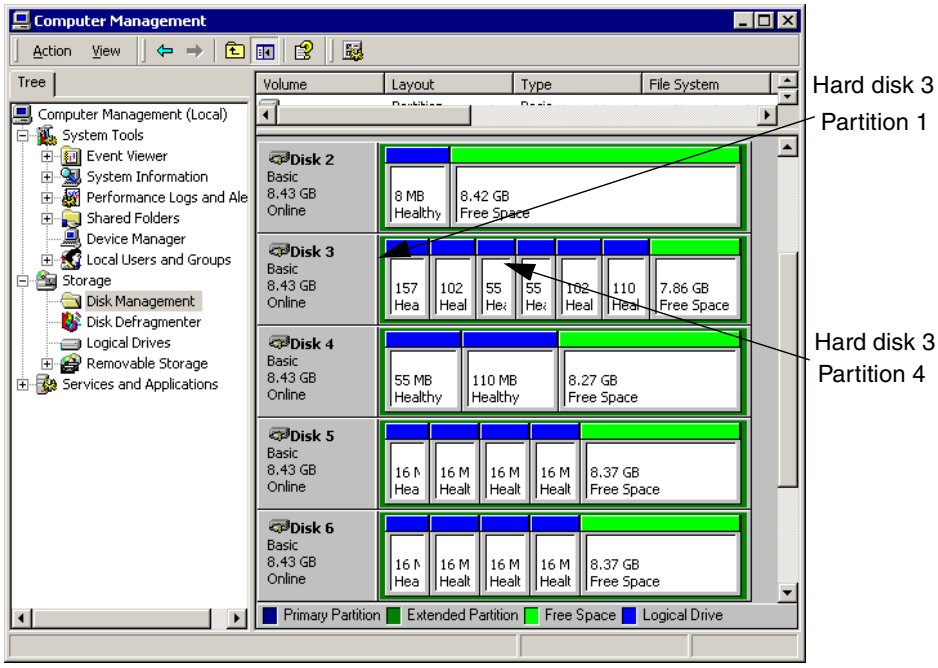


Figure 60. Hard disk and Partition number examples window

In Figure 61 is the .tbl file we created for node1 (itsoops1).

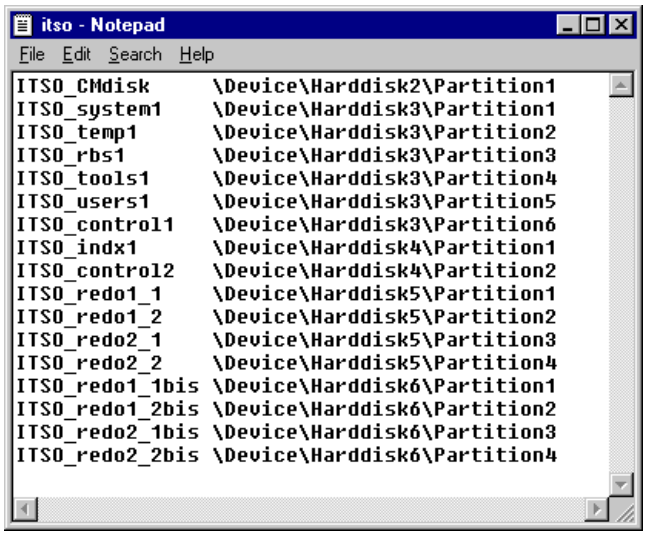


Figure 61. ITSO.tbl window

Note: Make sure you press Enter after the last line of the .tbl file.

2. Go in the directory where your .tbl file is located. Run the following command in order to map the drives to the symbolic names:

   ```
   setlinks /f:ITSO.tbl
   ```

3. Review the output window to make sure that the drives have been mapped with the correct names.

4. Copy the .tbl file on the second node, adapt it to the hard disks number value in step 1 on page 107 and repeat step 2 on the second node.

8.5 Creating a database script using Oracle DB Configuration Assistant

This section details the creation of a database using Oracle Database Configuration Assistant. We will use the configuration Assistant to create a batch file, then we will run the batch file:

1. From the Start menu select **Oracle Database Configuration Assistant**. This will open the configuration assistant. (See Figure 62.)

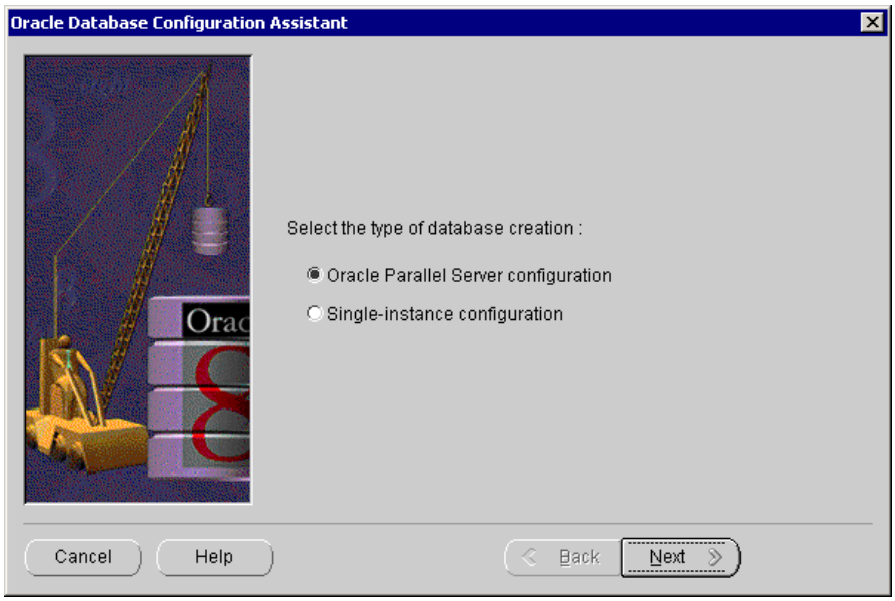


Figure 62. Oracle Database Configuration Assistant

1. Select **Oracle Parallel Server configuration**. Click **Next**. You will see a window similar to Figure 63.

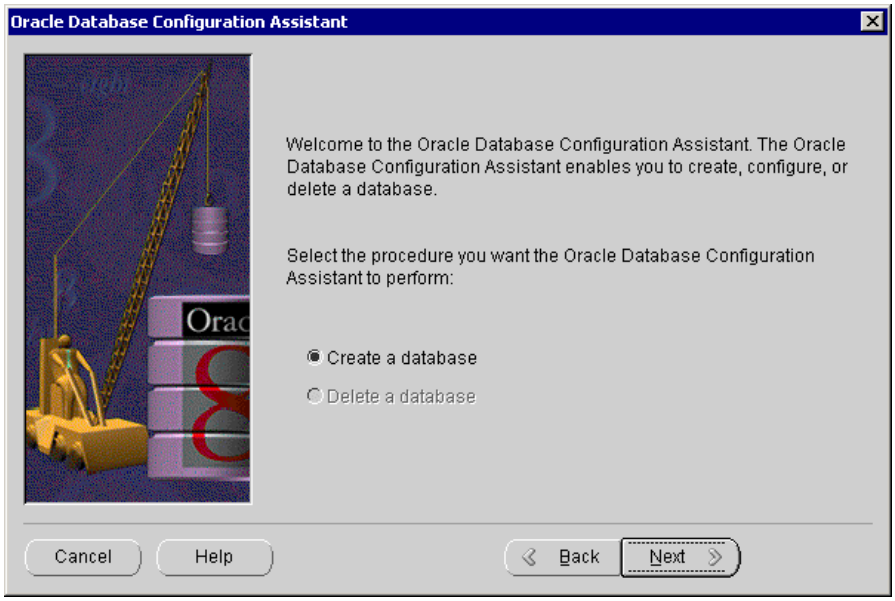


Figure 63. Oracle Database Configuration Assistant

2. Select **Create a database**, click **Next**. You will see a window similar to Figure 64.

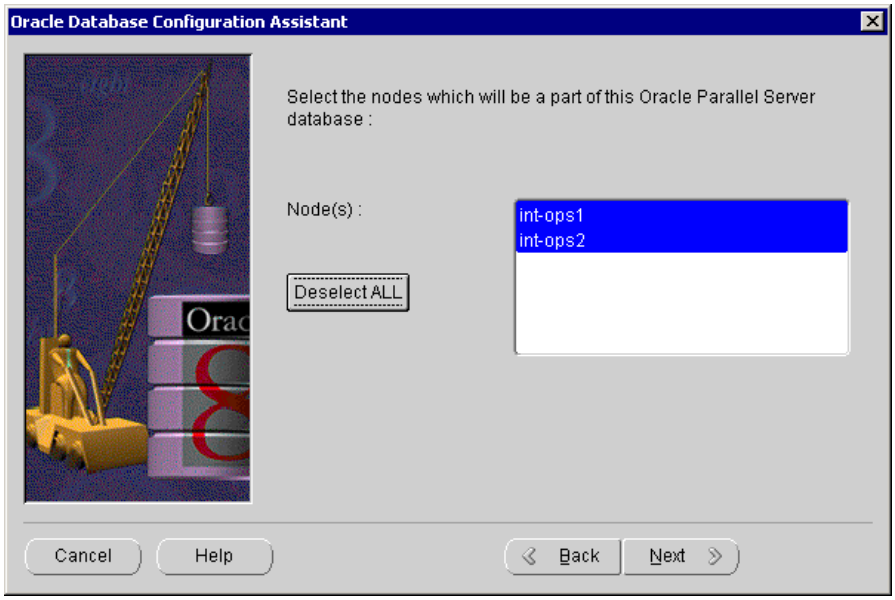


Figure 64. Oracle Database Configuration Assistant

3. Select all the nodes, click **Next**. You will see a window similar to Figure 65.

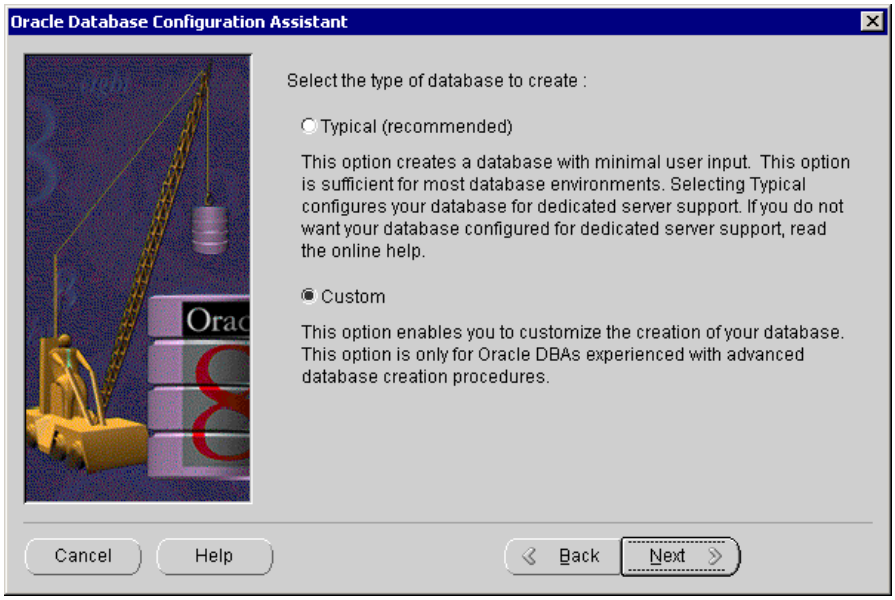


Figure 65. Oracle Database Configuration Assistant

4. Select **Custom** and click **Next**. You will see a window similar to Figure 5.

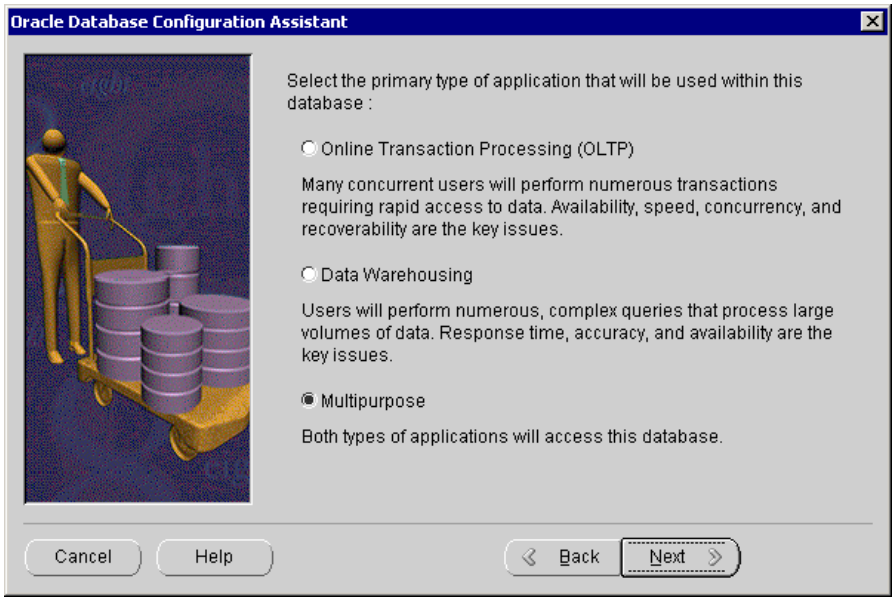


Figure 66. Oracle Database Configuration Assistant

5. This window enables you to choose what type of database you will be using. This can give certain characteristics to the way the database performs. Online Transaction Processing (OLTP), Data Warehousing, or a Multipurpose database will all perform slightly differently. Select **Multipurpose** and click **Next**. You will see a window similar to Figure 67.

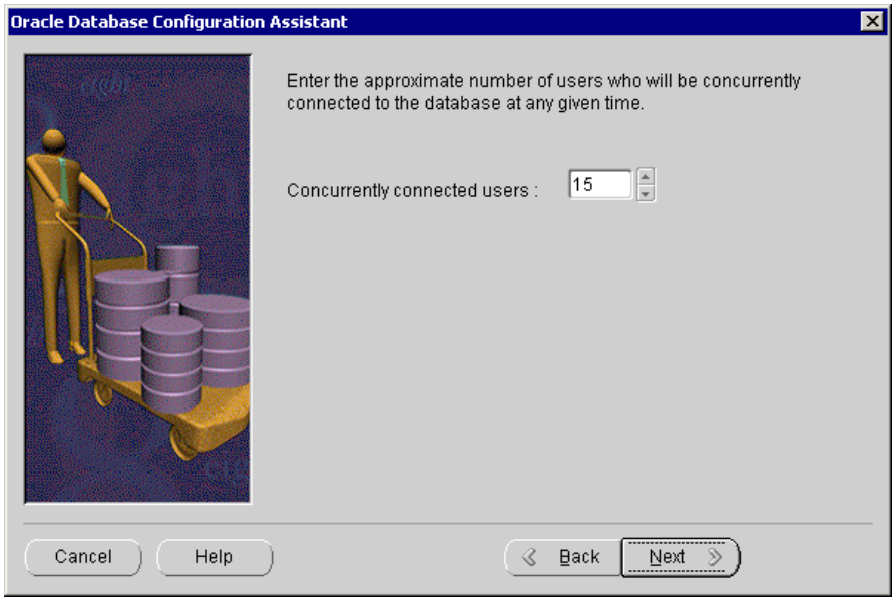


Figure 67. Oracle Database Configuration Assistant window

6. Enter the number of concurrent users in this database. For our example, we entered `15`. Click **Next**. You will see a window similar to Figure 68.

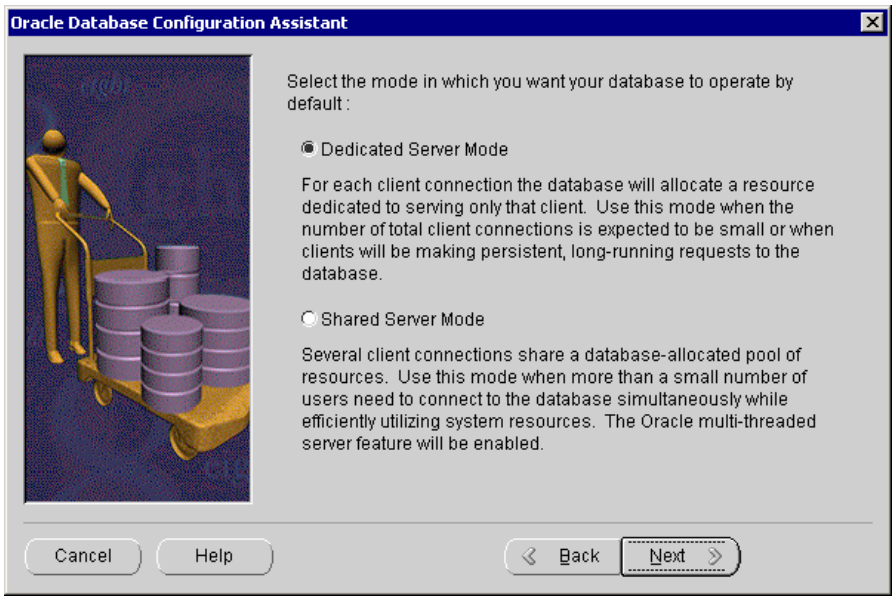


Figure 68. Oracle Database Configuration Assistant window

7. Next, you will select the operation mode for your database. Dedicated Server Mode is for a simple configuration with few users, while Shared Server Mode is for a more complex configuration. Select **Dedicated Server Mode**. In Chapter 9, “Load balancing and failover” on page 129, we will convert the database into a shared server mode for load balancing issues. Click **Next**. You will see a window similar to Figure 69.

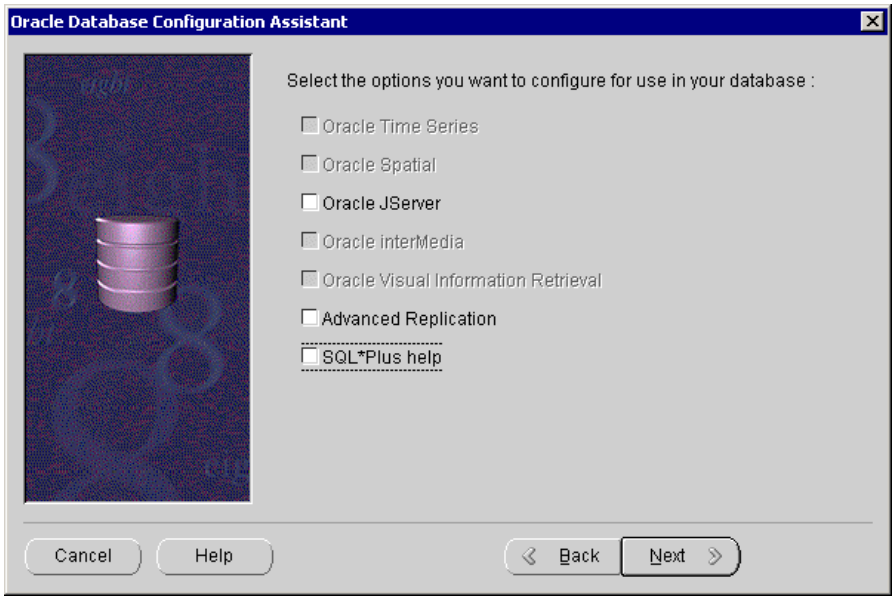


Figure 69. Oracle Database Configuration Assistant window

8. Check the option you need for your application. Also, we suggest you uncheck every option if you do not need them. Click **Next**. You will see a window similar to Figure 70.

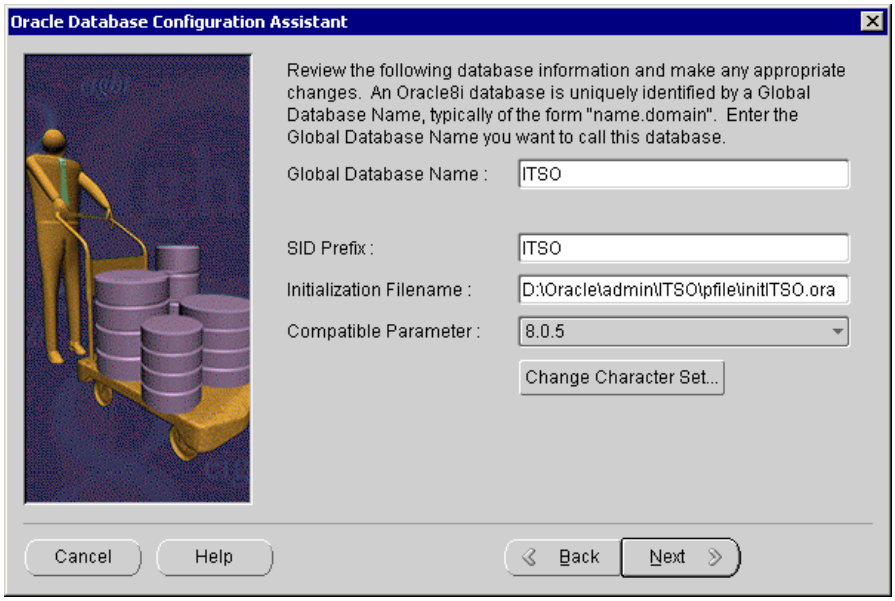


Figure 70. Oracle Database Configuration Assistant window

9. Enter the name of your database. For our example, we typed `ITSO`. Click **Next**. You will see a window similar to Figure 71.

   Note: If you use another name as your database name, make sure this is reflected throughout your configuration.

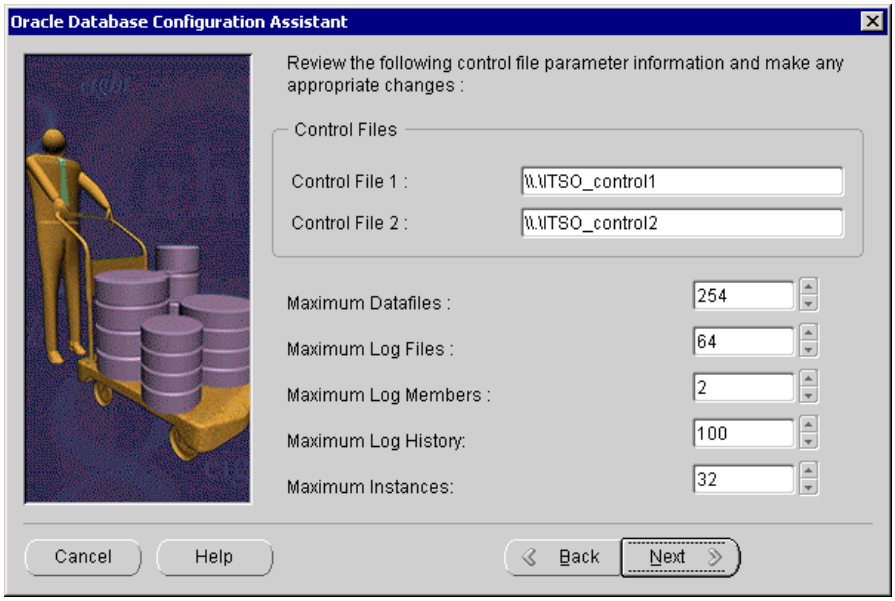


Figure 71. Control files parameters window

10. Enter the symbolic name of the two control file tablespaces as defined in Table 5 on page 87. Accept the default parameters. Click **Next**. You will see a window similar to Figure 72.

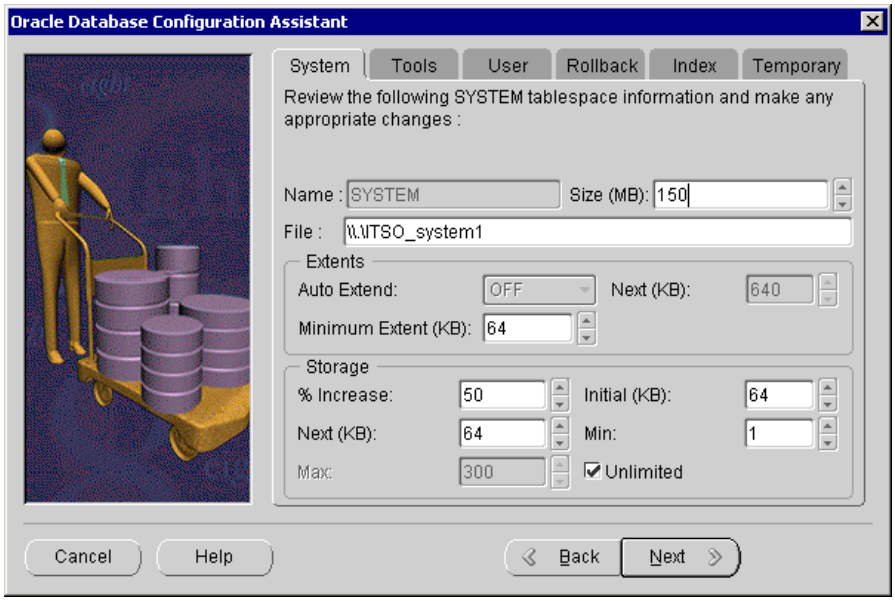


Figure 72. System tablespace information window

11. Enter the information for the System tablespaces, according to the database design you have created (see Table 5 on page 87):
    a. Set the tablespace size, for our example, `150`.
    b. In the File field, enter the symbolic name of the System tablespace. For example, \\.\ITSO_system1.

       Note: "\\.\" designates a symbolic name.

    c. Keep the default for the other values.
12. Repeat step 11 on page 117 for User, Rollback, Index, and Temporary tabs. Use the information from Table 5 on page 87.
13. Click **Next**. You will see a window similar to Figure 73.

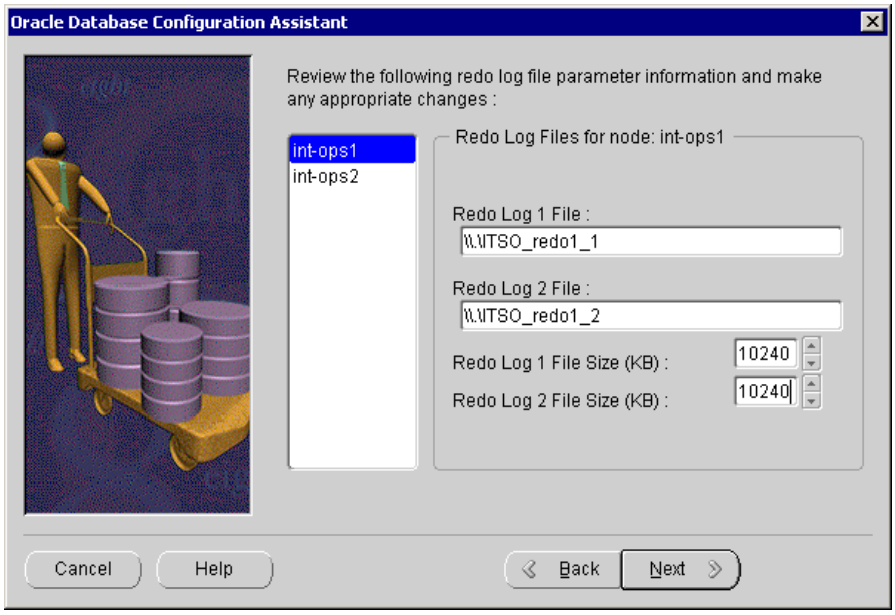


Figure 73. Redo log information window

14. Enter the information for the redo logs, according to the database design you have created (see Table 5 on page 87):

    In the Redo Log File fields, enter the symbolic name of the redo log tablespaces. For our example, int-ops1:

    - Redo Log 1 File = \\.\ITSO_redo1_1
    - Redo Log 2 File = \\.\ITSO_redo1_2

    For our example, int-ops2:

    - Redo Log 1 File = \\.\ITSO_redo2_1
    - Redo Log 2 File = \\.\ITSO_redo2_2

    Note: If additional nodes were a part of your cluster, you would repeat this step for all the nodes of the cluster by selecting the nodes in the list on the left of the window.

15. Once this task has been completed, click **Next**. You will see a window similar to Figure 74.

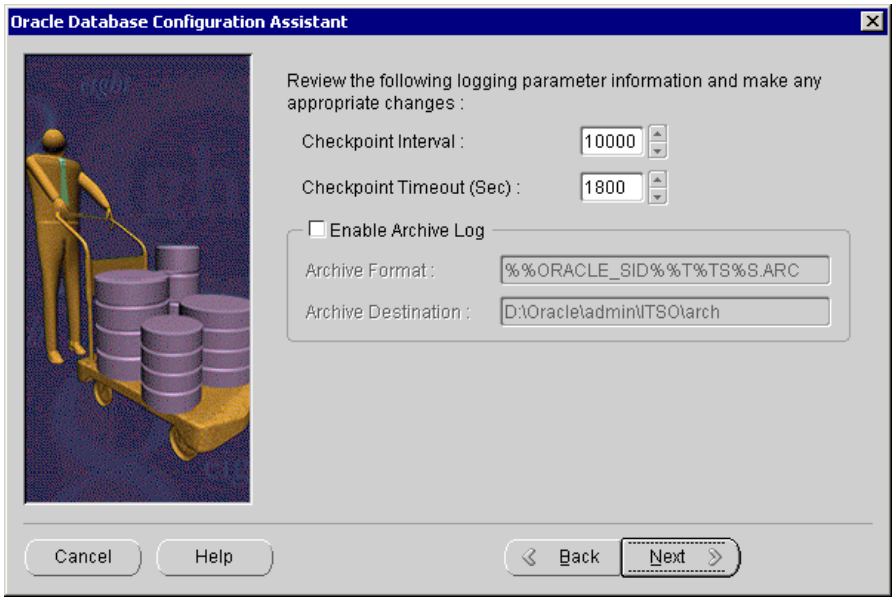


Figure 74. Checkpoint window

16. Accept the defaults and click **Next**. You will see a window similar to Figure 75.

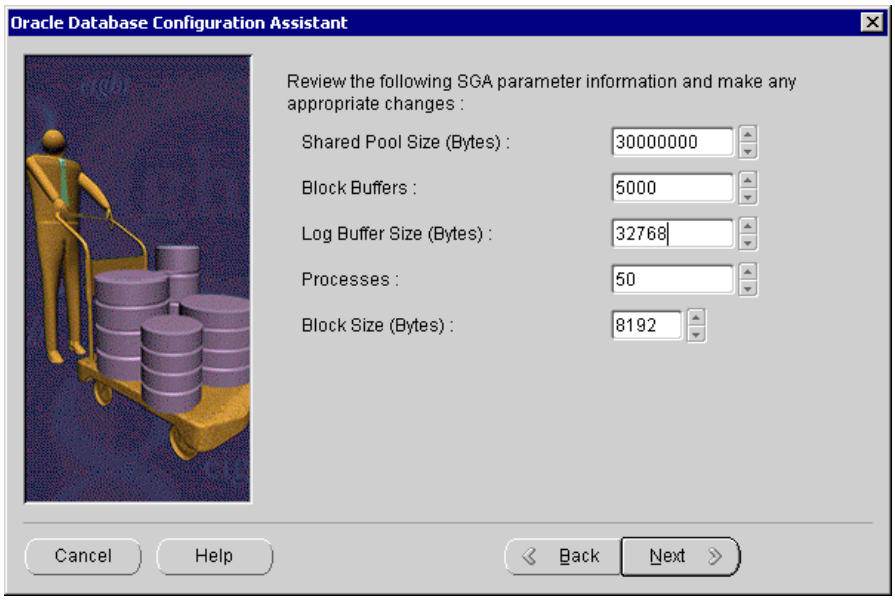


Figure 75. SGA parameter information window

17. Set the SGA values. Normally, we would accept the defaults and modify only the block size to reflect 8192 bytes. However, only for our test environment, we set the Shared Pool Size to 30000000 bytes and the Block Buffers to 5000 to generate a faster database startup.

    Note: We recommend you keep 8 KB block size. All the other values can be modified after the database has been created except for the block size.

18. Click **Next**. You will see a window similar to Figure 19.

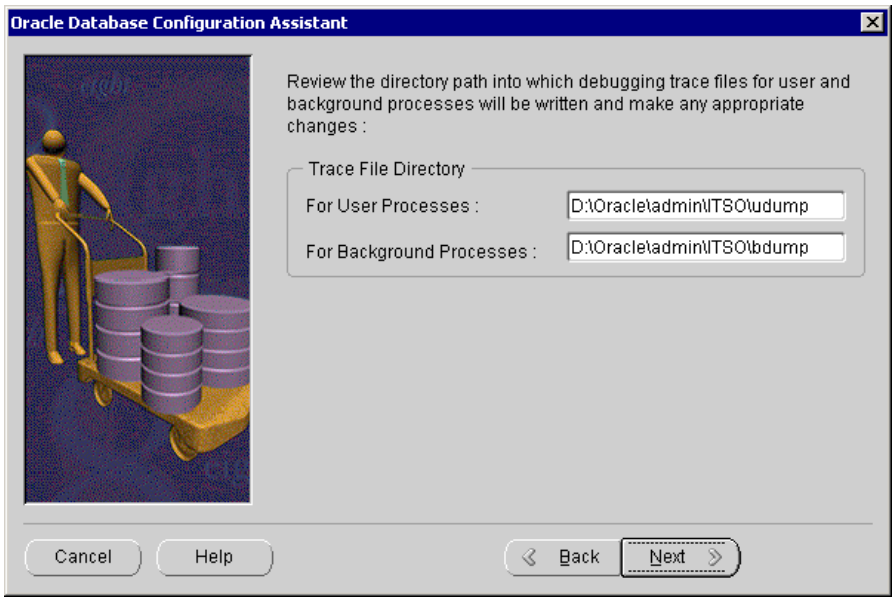


Figure 76. Trace File Directory path window

19. Accept the defaults and click **Next**. You will see a window similar to Figure 20.

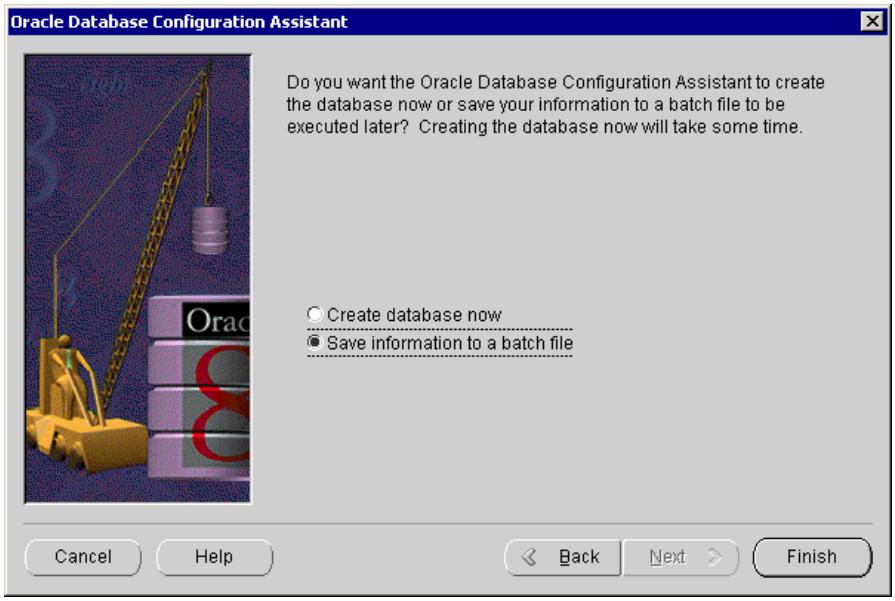


Figure 77. Create database or Save information window

20. Select **Save information to a batch file**. Save as createITSO.bat. The batch file will contain all routines required to create the database. This will allow you to make any adjustments required (for example, add member to the redo log groups) and re-run the batch file to re-create the database if you experience any problems. Click **Finish**.

21. Enter a file name and a location for the database creation script, for example, `ITSO1run.sql` and `D:\Oracle\Admin\createITSO`. Click **OK**.

22. Next, use Notepad to modify the database creation script to set up mirroring on the redo log files. Go to the directory where you saved the Database creation script (D:\Oracle\Admin\createITSO) and open it.

    a. Edit the file <your database>1run.sql (for example, ITSO1run.sql). Replace the original block with the modified block:

Original Block

```
LOGFILE
   '\\.\ITSO_redo1_1' SIZE 10240K REUSE,
   '\\.\ITSO_redo1_2' SIZE 10240K REUSE
MAXLOGFILES 64
MAXLOGMEMBERS 2
```

Modified Block

```
LOGFILE
    group 1 ('\\.\ITSO_redo1_1','\\.\ITSO_redo1_1bis') SIZE 10240K REUSE,
    group 2 ('\\.\ITSO_redo1_2','\\.\ITSO_redo1_2bis') SIZE 10240K REUSE
MAXLOGFILES 64
MAXLOGMEMBERS 4
```

b. Edit the file <your database>1pslog.sql (for example, ITSO1pslog.sql) Replace the original block with the modified block:

Original Block

```
alter database add logfile thread 2
 group 3 '\\.\ITSO_redo2_1' Size 10240K reuse,
 group 4 '\\.\ITSO_redo2_2' Size 10240K reuse;
```

Modified Block

```
alter database add logfile thread 2
   group 3 ('\\.\ITSO_redo2_1','\\.\ITSO_redo2_1bis') Size 10240K reuse,
   group 4 ('\\.\ITSO_redo2_2','\\.\ITSO_redo2_2bis') Size 10240K reuse;
```

Note: These modifications are fit for a two-node cluster. Adapt them if you have more than two nodes in your cluster. A thread is a node for Oracle.

8.6 Creating the database

The process of creating the database will take several minutes.

1. Make sure the Oracle CM service is started on all the nodes.
2. Open a DOS command prompt on the node where you have created the database creation script, go to the script location, and run the database creation script:

   Type `creatITSO.bat` and press Enter. Wait until the creation finishes.
3. Shut down the database:
   a. At the DOS prompt type `svrmgrl` and press Enter:

      ```
      c:\> svrmgrl
      ```

      This will load the Oracle Server Manager (OSM).
   b. At the OSM prompt, type `connect internal/oracle` and press Enter:

      ```
      SVRMGR>connect internal/oracle
      ```

c. Type `shutdown` and press Enter:

```
SVRMGR>shutdown
```

8.7 Validation

This section describes the validation of the Oracle Parallel Server using the database that was created.

8.7.1 Starting services

In this section, you will start the OracleService<DB name>*x* instance on each node (where *x* is the node number, DB name is your database name).

From each node go to the control panel, select **Services**, select **OracleServiceITSO1** and click **Start**.

Hint: This can also be achieved through a command prompt, for example:

```
c:\> NET START OracleServiceITSO1
```

where `ITSO` is the name of the database.

8.7.2 Starting the database in parallel mode

Carry out the following procedure on each node individually:

1. Open an MS-DOS command line, run the Oracle Server Manager (OSM):

   ```
   c:\> svrmgrl
   ```

2. At the OSM prompt type, type `connect internal/oracle`, press Enter:

   ```
   SVRMGR>connect internal/oracle
   ```

3. At the OSM prompt type:

   ```
   SVRMGR> startup pfile=d:\oracle\admin\op\pfile\initITSOX.ora
   ```

This will initialize the ITSO database. **X** is the node number. For example, the init.ora file for node1 is called initITSO1.ora and for node 2 the file is called initITSO2.ora.

If you get an error message telling you that you have run out of memory, modify the SGA in the init.ora file:

4. Perform steps 1 through 3 on all nodes using the appropriate init<DB name>**X**.ora file specified in step 3.

8.7.3 Verifying that instances are running on a node

Type the following at the SVRMGR> prompt:

```
SVRMGR> select * from v$active_instances;
```

This will display all nodes that have had the database instance successfully installed.

```
LSNRCTL> start LISTENER
Starting tnslsnr: please wait...

Failed to open service <OracleOraHome81TNSListener>, error 1060.
TNSLSNR for 32-bit Windows: Version 8.1.6.0.0 - Production
System parameter file is D:\Oracle\Ora81\network\admin\listener.ora
Log messages written to D:\Oracle\Ora81\network\log\listener.log
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\EXTPROC0ipc
)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=itsoops2.headquarters.wo
rld.com)(PORT=1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC0)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for 32-bit Windows: Version 8.1.6.0.0 - Produc
tion
Start Date                24-JUL-2000 23:54:13
Uptime                    0 days 0 hr. 0 min. 0 sec
Trace Level               off
Security                  OFF
SNMP                      OFF
Listener Parameter File   D:\Oracle\Ora81\network\admin\listener.ora
Listener Log File         D:\Oracle\Ora81\network\log\listener.log
Services Summary...
  PLSExtProc            has 1 service handler(s)
The command completed successfully
LSNRCTL> exit
```

Figure 78. Startup and shutdown process, and running instances check

8.8 Modifying Oracle services startup

We have seen that certain services within Windows 2000 are required in order for Oracle8*i* R2 and the OPS components and databases to function correctly.

8.8.1 Creating service dependencies

Problems may arise when services, such as instances, try to start automatically before the OSD service (OracleCMService) has started.

An example of this is the database instance service OracleServiceITSO1 (the service used for the database we created earlier). If this service attempts to load before OracleCMService (OSD) then it will fail and the instance will not be able to run on that node.

There are two options you should consider to avoid startup problems:

1. Make the OSD autostart and the OPS service manual start.
2. This option requires you to edit the Registry. Unless you have experience in editing the Registry, you should use option one.

   It is possible to set the OracleServiceop1 to start the service manually upon each reboot, but the preferred method is to set a dependency on this service. This will ensure that the service is automatically started *after* OracleCMService. This section describes the dependency creation.

   a. Open the registry editor (regedt32).

   b. Select HKEYLOCALMACHINE, SYSTEM, CurrentControlSet, Services and OracleServiceop1 for node1.

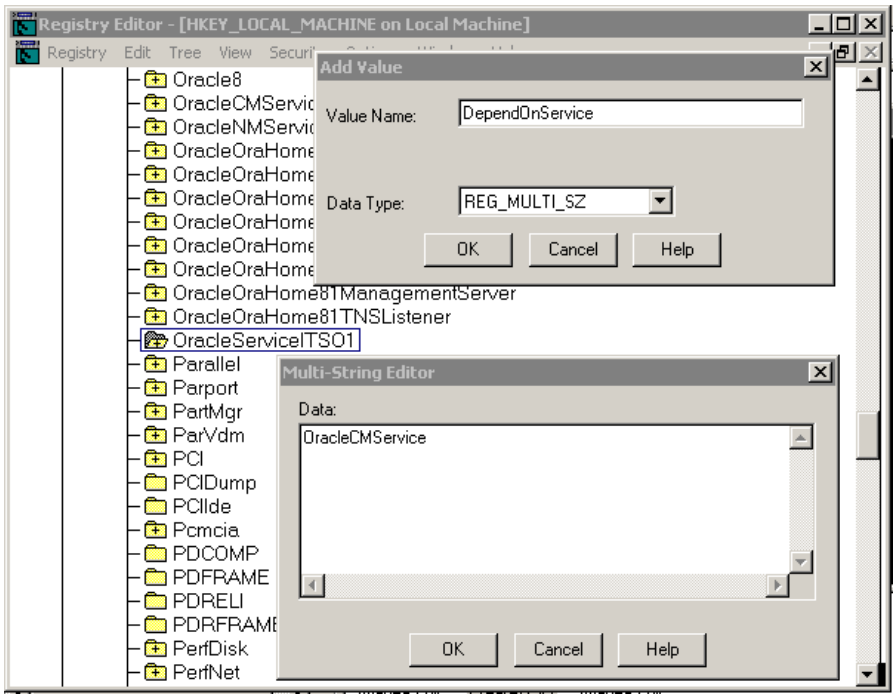


Figure 79. Creation of dependency on OracleCMService

   c. Select **Add Value**. Enter the Value Name: “DependOnService”

d. Choose REG_MULTI_SZ as the data type.

e. Type OracleCMService in the Multi-String Editor and click **OK**.

f. Check the registry entry to ensure that the dependency has been successfully saved (DependOnService: REG_MULTI_SZ: OracleCMService).

g. Perform steps a through f on all nodes. Restart the cluster and check that the required services have started successfully.

8.8.2 Automatic startup

For an easier administration, make these services automatically started, as shown in Figure 80.

- OracleCMService
- OracleServiceITSO1 (ITSO is the database, 1 for node 1)
- OracleOraHome81TNSListener (it should already be automatic)

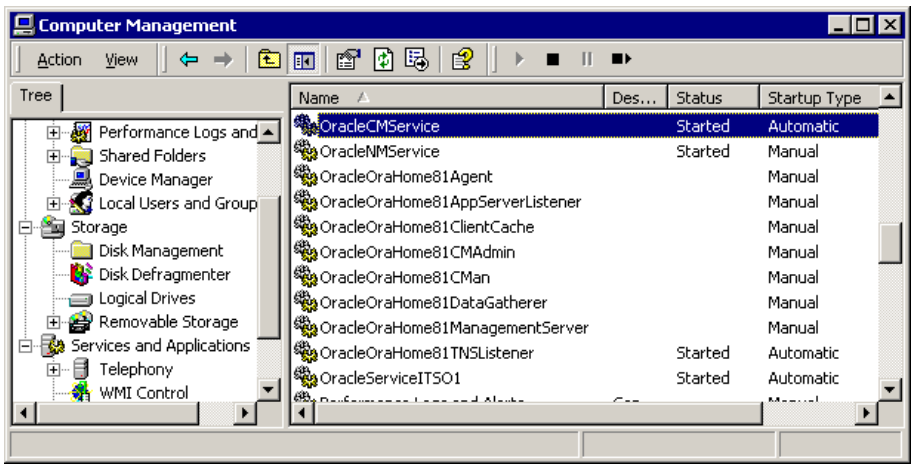


Figure 80. Oracle Services window

To change the Startup type, perform the following steps:

1. Right-click the service, select **Properties**, change the Startup type to Automatic.

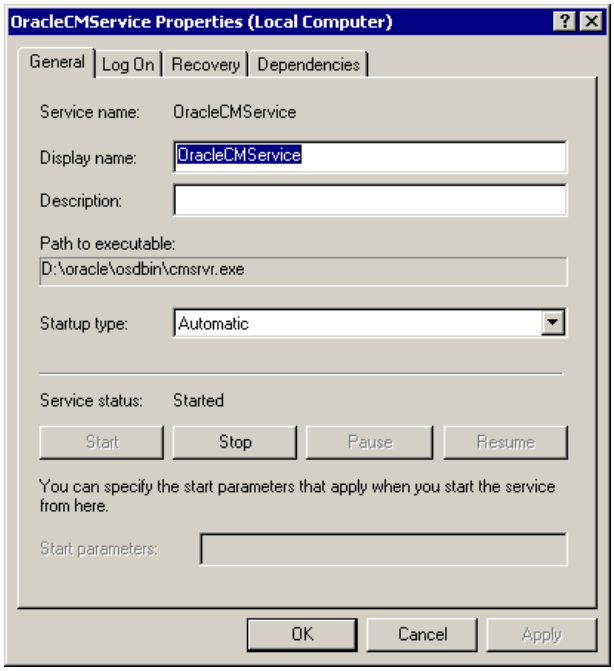


Figure 81. OracleCMService Properties for automatic startup window

2. Click **OK**.

Chapter 9. Load balancing and failover

In this chapter, we will see how to configure load balancing and the failover on an OPS cluster.

9.1 Hardware availability

The hardware high availability is provided by a fully redundant Fiber Channel architecture. In case of a component failure, the system can use a different path to access the disks. The RDAC driver enables the system to do this.

If you have a Fiber Channel adapter in a server, you have to reinstall the RDAC driver.

9.2 Client load balancing and failover

A connection to a single instance database is made to the instance (the SID). With OPS, the connection is made to a service, it represents the database from whichever instance.

Check the INIT.ORA file from each instance, and you will see that the database name and a service name is defined, and a different instance name is defined for each node.

- db_name = "ITSO"
- service_names = ITSO
- instance_name = ITSO1

The configuration of the connection between the client and the server is done through two files, the LISTENER.ORA for the server side, and the TNSNAMES.ORA for the client side.

9.2.1 Listener configuration

The LISTENER.ORA, located in D:\Oracle\Ora81\Network\Admin (depending on where you have made your Oracle installation) needs few modification. The Database Configuration Assistant has set up it with the interconnect host name (for example, int-ops1). Change this host name to the LAN host name for the node (for example, ITSOOPS1). Check that the LISTENER.ORA looks like Figure 82.

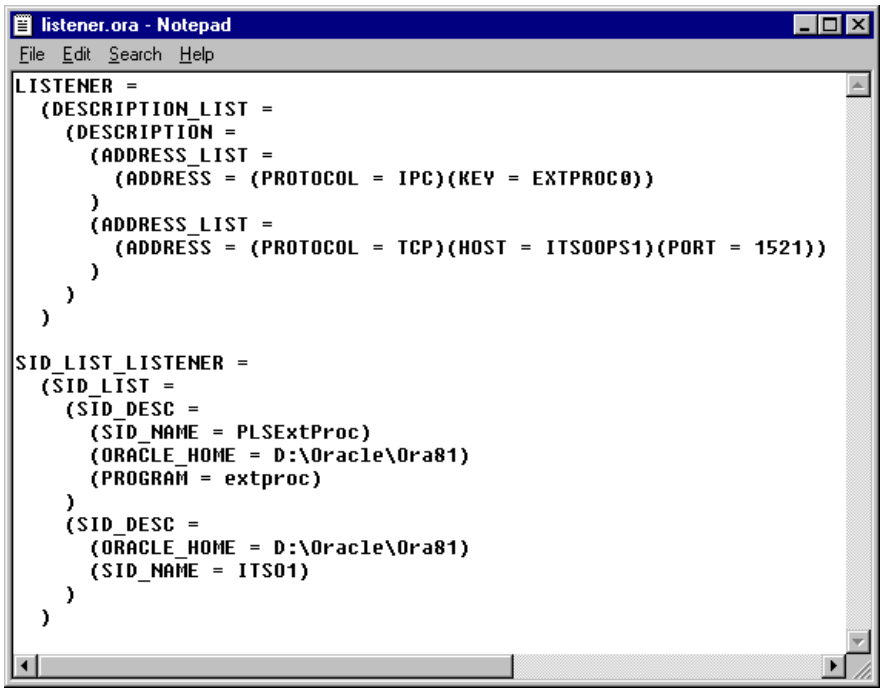


Figure 82. Listener.ora

Note: The LISTENER.ORA is from the node ITSOOPS1; therefore the SID_NAME is set to the instance ITSO1. For the node ITSOOPS2, its SID_NAME is set to the instance ITSO2.

You have to restart the listener on both nodes. At the DOS prompt, run the following commands:

```
lsnrctl stop

lsnrctl start
```

The connection will be done as shown in Figure 83 on page 131. The configuration of the tnsnames.ora will determine to which server listener that the client will connect.

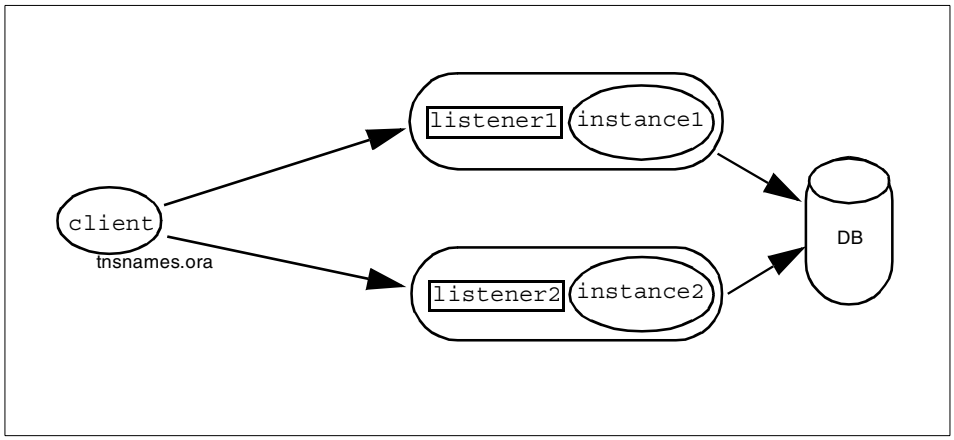


Figure 83. Client connection to any server listener

9.2.2 TNSNAME.ORA configuration

The TNSNAMES.ORA is located in D:\Oracle\Ora81\Network\Admin (D:\Oracle\Ora81 depends on where you have made your Oracle installation). The Database Configuration Assistant has set it up with the interconnect host name (for example, int-ops1 and int-ops2). Change this host name with the LAN host name for the nodes (for example, ITSOOPS1 and ITSOOPS2).

The TNSNAMES.ORA has the client load balancing enabled by default. Each entry in this file corresponds to a different type of connection:

- ITSO.HEADQUARTER.WORLD.COM represents the database service ITSO. It does load balancing and failover between the two nodes ITSOOPS1 and ITSOOPS2.
- ITSO1.HEADQUARTER.WORLD.COM represents the instance on ITSOOPS1. Any connection using this entry will be made to ITSO1 (because INSTANCE_NAME is specified).
- ITSO2.HEADQUARTER.WORLD.COM represents the instance on ITSOOPS2. Any connection using this entry will be made to ITSO2.
- ITSO1_STARTUP.HEADQUARTER.WORLD.COM and ITSO2_STARTUP.HEADQUARTER.WORLD.COM are entries to connect to the instances in dedicated mode with sqlplus for administrative task (useless if you use SVRMGRL).

We recommend you change the line:

```
(LOAD_BALANCE = yes)
```

to

```
(LOAD_BALANCE = ON)
```

The LOAD_BALANCE parameter specifies that the client request will be randomized among multiple listeners.

The FAILOVER parameter specifies that the request fail over to the next listener in the list until the connection is successful.

Set these parameters to ON or OFF if you want to enable or disable them.

See the TNSNAMES.ORA in Figure 84.

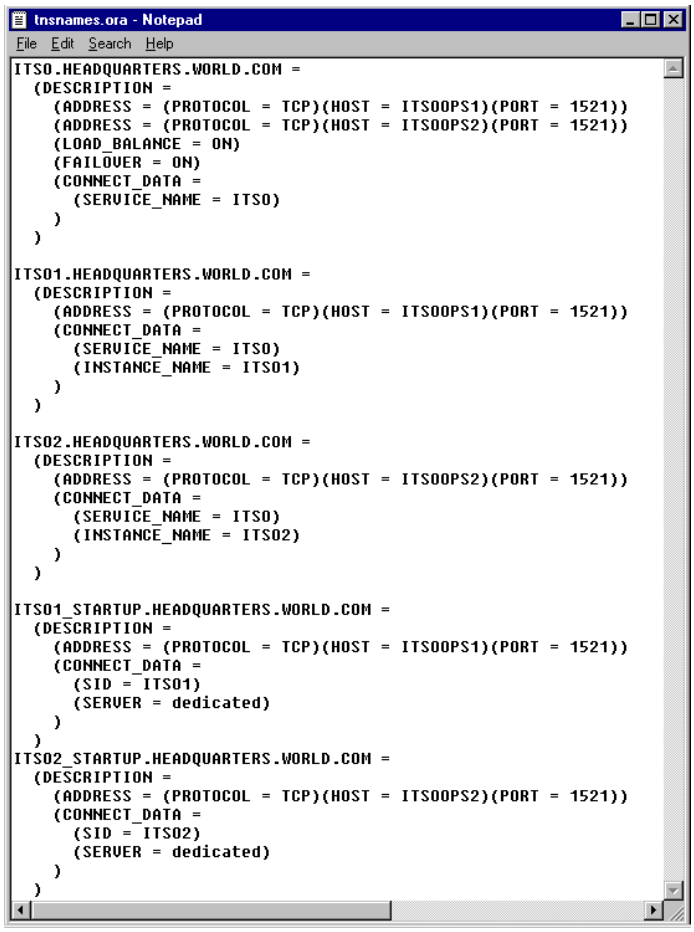


Figure 84. TNSNAME.ORA configured for client load balancing and failover

By connecting to the service ITSO, with the command

```
SQLPLUS system/manager@ITSO
```

you will be connected to either ITSOOPS1 or ITSOOPS2.

9.3 Connection load balancing

For the creation of the database, in the Database Configuration Assistant in 8.6, “Creating the database” on page 122, we have chosen the Dedicated Server Process mode as the database operational mode. That is the basic mode.

The other mode, the Multi-Threaded Server (MTS) eliminates the need for a dedicated server process for each connection. A small number of shared servers can perform the same amount of processing as many dedicated servers. Therefore, a database with the MTS mode can handle many more user connections. Plus, the MTS mode offers the capability of load balancing based on the server activity and the number of clients connected.

The connections are routed via a dispatcher.

When MTS is implemented, the instances automatically register with the listeners from the different nodes of the cluster. This registration provides the listener with the service names, instance names, and network addresses of all instances. The different listeners will be able to establish the connections to the registered instances using load balancing based on the servers’ CPU load and the number of connections (this information is provided by the PMON process).

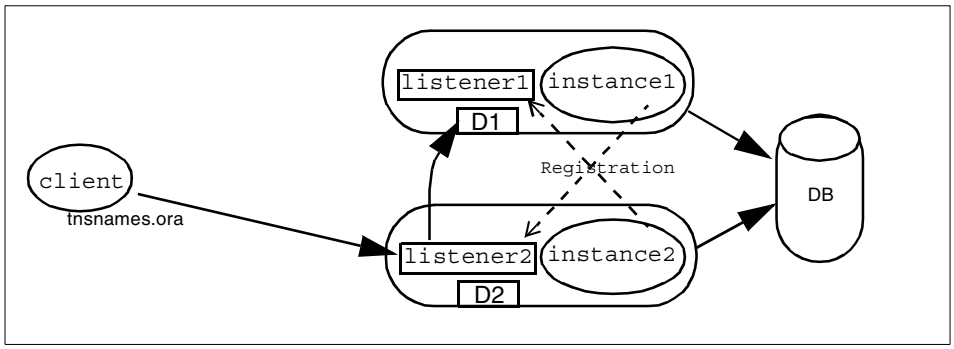


Figure 85. MTS model client connection

In Figure 85 is an example of a connection when the MTS is implemented. The client asks for a connection to the listener 2 (which is running on node 2). The listener 2 has two instances registered (instance1 and instance2). The load and the number of connections on instance1 are lower than on instance2. So listener 2 establishes the client connection to the dispatcher D1 from node1.

To implement this, two tasks have to be done:

1. Add a list of the different listeners to the bottom of the TNSNAMES.ORA of each node. This is an example of the list to add:

```
ops_list.HEADQUARTERS.WORLD.COM =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = ITSOOPS1)(PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP)(HOST = ITSOOPS2)(PORT = 1521))
  )
```

   Make sure you use the correct host name and the correct domain (refer to the other entries in your TNSNAMES.ORA).

2. Add the MTS parameter to the INIT.ORA of each node. This is the line to add (to InitITSO.ORA for example, see Figure 86):

```
MTS_DISPATCHERS="(PROTOCOL=TCP)(listener=ops_list.HEADQUARTERS.
WORLD.COM)"
```

   Make sure you use the correct domain.

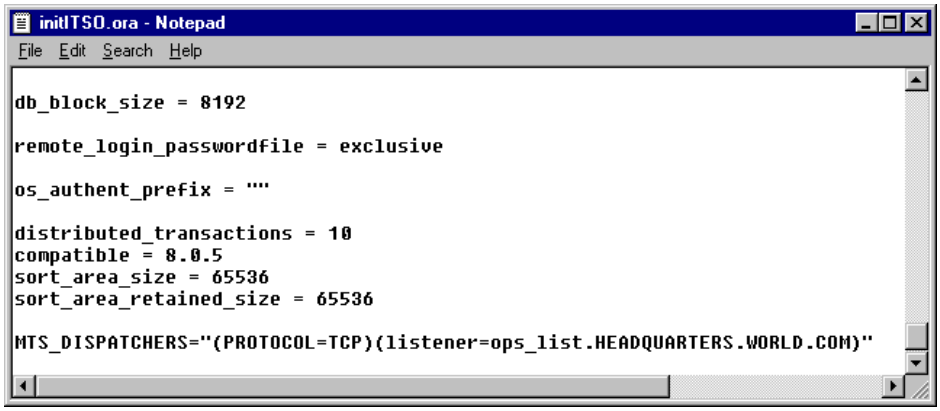


Figure 86. InitITSO.ORA

Note: Client load balancing and failover (9.2, “Client load balancing and failover” on page 129) and connection load balancing (MTS) are

complementary. Listener load balancing and high availability is provided by client load balancing and failover). The instance load balancing and high availability is provided by the MTS model.

These are useful views:

- The view v$instance gives the name of the instance and the hosts on which you are connected.
- The view v$dispatcher gives information about the dispatchers (number of connections being handled by each dispatcher).
- The view v$dispatcher_rate gives statistics about the rate at which a dispatcher is receiving and handling messages, events, etc.
- The view v$circuit gives information on the different connections to the instance dispatchers.

Run queries on these views from a SQL Plus session (see Figure 87). For v$circuit, run the query from SVRMGRL, because SVRMGRL uses a dedicated connection. It is not connected to a dispatcher, so it will not interfere with the other connections.

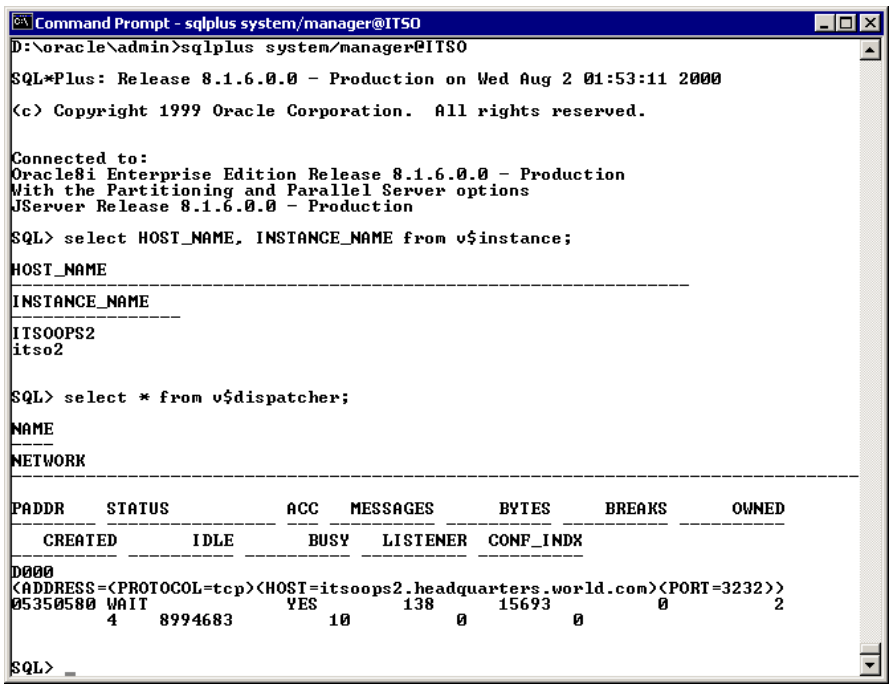


Figure 87. Queries on the v$instance, and v$dispatcher

Note: For more information on the MTS and the configuration of the dispatchers, see:

- *Configuring Multi-Threaded Server, Net8 Administrator's Guide, Release 8.1.6*
- *Configuring Dispatchers, Oracle8i Tuning, Release 8.1.6*

If one instance fails, the connections established to this instance are lost and the users have to reconnect manually. We will see in the next section how to have a transparent failover.

9.4 Transparent application failover

With ttransparent application failover (TAF), the users are not affected by on an instance crash. If one instance crash, the users session is handled by another node. To have this kind of failover available, the application that connects to the instances must use the Oracle Call Interface (OCI) for the communication (not ODBC/JDBC). SQL*PLUS, for example, is using the OCI.

To enable this, a clause must be added in the client's TNSNAMES.ORA. This is the model of this clause:

```
(FAILOVER_MODE=
    (BACKUP=<backup node>)
    (TYPE=<type>)
    (METHOD=<method>)
)
```

Options for the failover TYPE are:

- NONE: No failover is attempted,
- SESSION: Following the failover, the interrupted activity will not be resumed,
- SELECT: An interrupted query will restart to provide missing records from the result set.

Options for the METHOD are:

- BASIC: The failover involves starting a new process for the user on the backup node,
- PRECONNECT: The backup node is already running a shadow process, and the user attaches to that process during failover.

This clause has to be added to the CONNECT_DATA clause of a listener service description.

We have enabled this failover with the TYPE set to SELECT, and METHOD set to PRECONNECT in the ITSO description. The preconnect method takes a backup connection besides the working connection. If you have 1000 users connected to a cluster, with the preconnect method you will have 2000 connections. These connections consume memory, and most of the time for nothing.

Note: We still recommend using PRECONNECT to implement a satisfactory transparent application failover.

In our configuration, the three levels of load balancing and failover are implemented (client load balancing and failover, connection load balancing, and TAF). With this configuration, we get the maximum availability for the cluster:

- If a listener fails, the requests go to the other listener (client load balancing and failover)
- If an instance fails, the requests go to the other instance (connection load balancing)
- If the connections are not interrupted by a failover, they are redirected to another instance (TAF)

In Figure 88 is the TNSNAMES.ORA configured for the three levels of load balancing and failover.

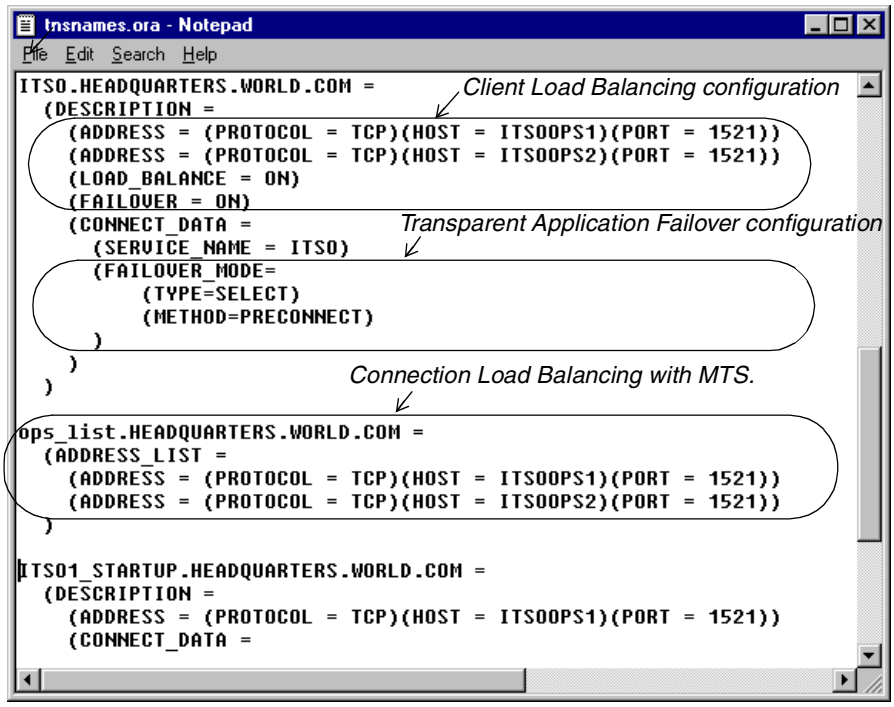


Figure 88. TNSNAMES.ORA for TAF

9.5 Client deployment

If there is no DNS, each client must have the /etc/hosts configured with the node's IP address, the node's name, and the node's name with domain (6.1.1, "Preparation - all nodes" on page 57).

The clients must have a TNSNAMES.ORA configured for load balancing and failover. We recommend you put the ITSO, ITSO1, ITSO2 service name entries (see Figure 89), do not put the ops_list entry we have put on the servers. Adapt the entries with the choice you make for the TAF strategy. The ITSO1 and ITSO2 service name entries will be useful to connect clients to a specific instances. For example, if you start an SQL*PLUS connection using the command:

```
sqlplus system/manager@ITSO1
```

you will be connected to the instance ITSO1.

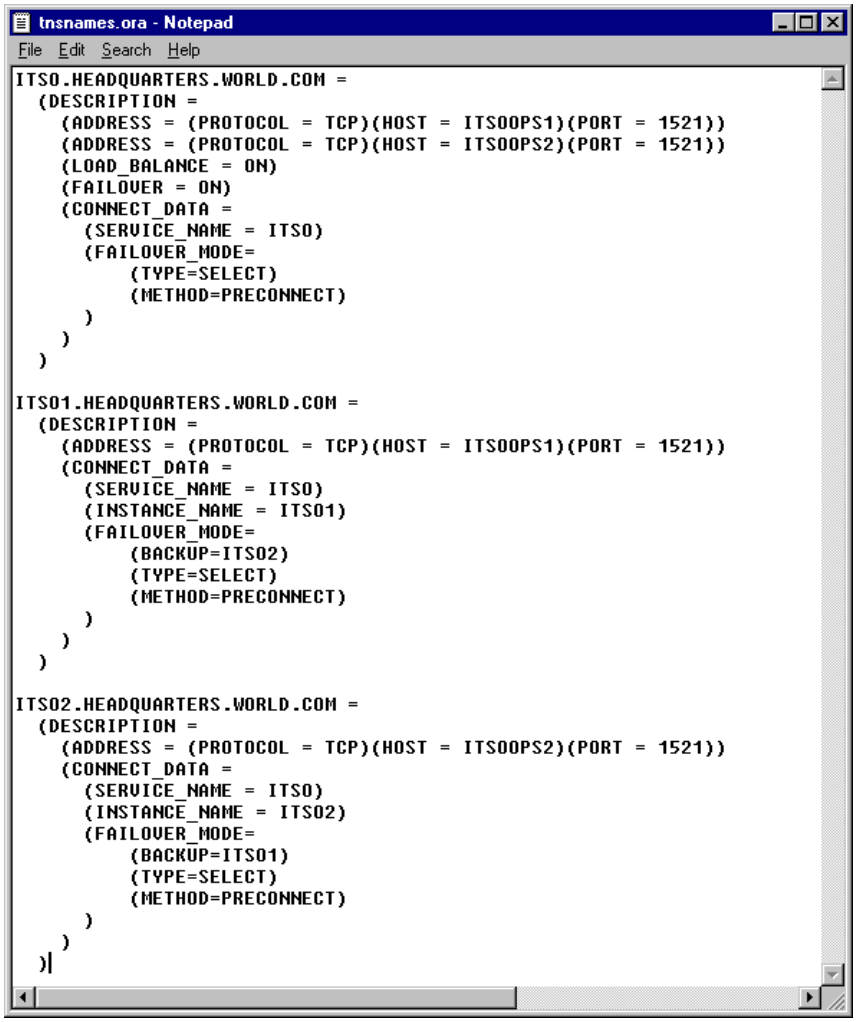


Figure 89. Client TNSNAMES.ORA

Chapter 10. Database migration

This chapter shows a simple way to migrate your data from a previous production database to your new OPS system. This simple way consists of exporting the database content to files, copy the files to the new system and importing the database content into the new database.

10.1 Definition

For this task, we will use the Export and Import utilities from Oracle.

The following export and import definition is from the Oracle8*i* Utilities manual:

> "The Export and Import utilities provide a simple way for you to transfer data objects between Oracle databases, even if they reside on platforms with different hardware and software configurations. Export extracts the object definitions and table data from an Oracle database and stores them in an Oracle binary-format Export dump file located typically on disk or tape.
>
> Such files can then be FTPed or physically transported (in the case of tape) to a different site and used, with the Import utility, to transfer data between databases that are on machines not connected via a network or as backups in addition to normal backup procedures."

10.2 Preliminary steps

Listed are some preliminary steps you should consider for database migration:

- You will need a network between the old system and the new system, or a tape drive on each system, to transfer the files.
- To process the migration, the new database must be up and running. It must contain the same tablespaces with the same file size or larger.
- To list the datafiles names, sizes, and associated tablespaces execute from SQL*Plus the following query on the old database:

  SELECT file_name, bytes, tablespace_name

  FROM sys.dba_data_files;

To modify a tablespace, use the ALTER TABLESPACE SQL statement (see the Oracle8*i* SQL documentation). You can add a raw partition (Oracle sees

the raw partition as a file) to a tablespace to give the tablespace more disk space. To add a raw partition, follow the process from 8.2, “Create RAID arrays” on page 82 and 8.3, “Implementing raw partitions” on page 98, modify the TBL file, and reload it with setlinks.

Make sure the old production database is not accessible by the users or read only. It must not be modified during the migration, or you would lose data.

Prepare free disk space for your export file.

10.3 Export step

To run a full export of the old production database, run this command at a DOS prompt from the old production server:

```
EXP system/manager@PROD full=Y file=d:\PROD.dmp filesize=500M
```

- `manager` is your system user password
- `PROD` is the service/instance name of the old production database
- `d:\PROD.dmp` is the file and the location in which you want to export the database
- `Filesize` is the available free space you have in the drive to which you are going to export the data

Note: If the export reaches the maximum file size, it will prompt you for another filename. This will allow you to save the export file just created onto a tape, or on the new server through the network, and free some disk space to continue the export. You can start the import process on the new server with the export file that has been created, and continue later on with the other export files.

10.4 Import step

To run the import of the database, run this command at a DOS prompt from one of the nodes (for example the one that hosts ITSO1 instance):

```
IMP system/manager@ITSO1 full=Y file=d:\PROD.dmp filesize=500M
```

- `manager` is your system user password
- `ITSO1` is the instance name of the node on which you have copied the export file.
- `d:\PROD.dmp` is the export file and its location

10.5 Invalid objects

You may have invalid objects after the import on the new database.

Run this query from a SQL*PLUS session to list the invalid objects:

```
Select object_name, object_type, owner, status from all_objects where
status='INVALID';
```

If some rows are returned, you have invalid objects. Check the Object_type and Owner (schema) column of the previous query.

If the invalid objects are packages, re-compile them running the following query from a SQL*PLUS session:

```
Alter package schema.XXX compile;
```

If the invalid objects are views, re-compile them running the following query from a SQL*PLUS session:

```
Alter view schema.XXX compile.
```

If some invalid objects are triggers, recompile them running the following query from a SQL*PLUS session:

```
Alter trigger schema.XXX compile;
```

Chapter 11. Oracle Enterprise Manager

In this chapter we discuss Oracle Enterprise Manager (OEM). First we provide an overview of Oracle Enterprise Manager, the requirements for installing the product and the benefits of using Oracle Enterprise Manager. Then we go through the installation procedures.

The Oracle Enterprise Manager installation is optional. It is not required for an OPS cluster. OEM provides graphical administration tools for the Oracle databases.

11.1 Overview

Oracle Enterprise Manager has a three-tiered architecture (Figure 90). The first tier is the client or administration console, which consists of a Java-based console and integrated applications. The second tier is the management server that provides control between the clients (first tier) and the different components that will be managed (third tier). The third tier can, for example, be an OPS solution running, for example, a parallel database, but it can also be normal databases or other services.

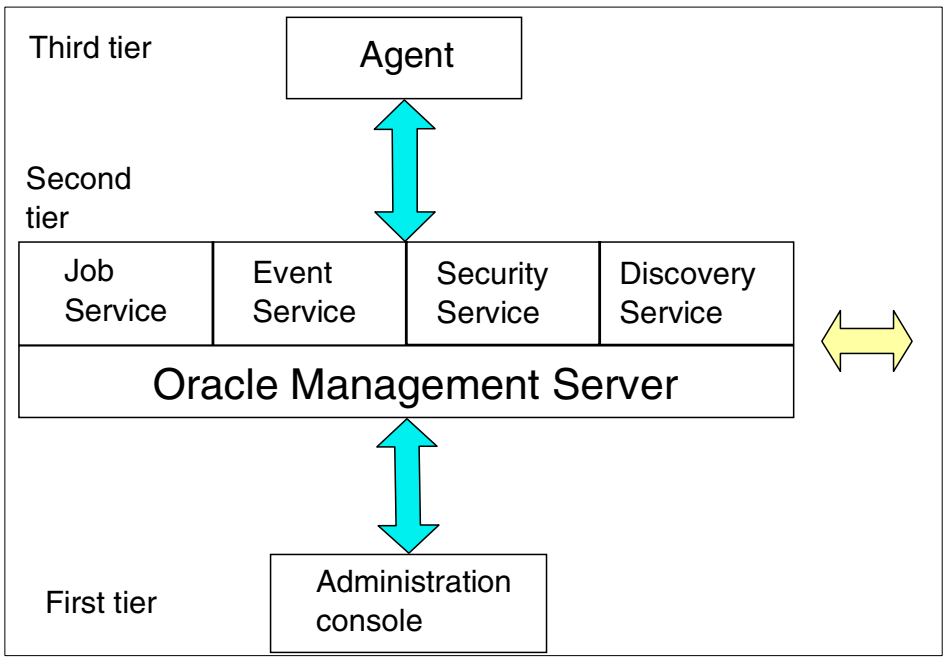


Figure 90. Oracle Enterprise Manager architecture

11.1.1 Definition

Oracle Enterprise Manager allows you to manage, administer, and monitor multiple complex databases from a single workstation.

Oracle Enterprise Manager is a set of centralized services and management applications that provide database administrator (DBA) for all the tools to manage multiple instances and parallel databases.

11.1.2 First tier

The first tier is the client or the administration console. The Oracle Enterprise Manager console is a client GUI that provides menus, drawers and tool bars. The console consists of four windows (see Figure 91 on page 147) for different purposes:

- Navigation window
- Group window
- Jobs window
- Event window

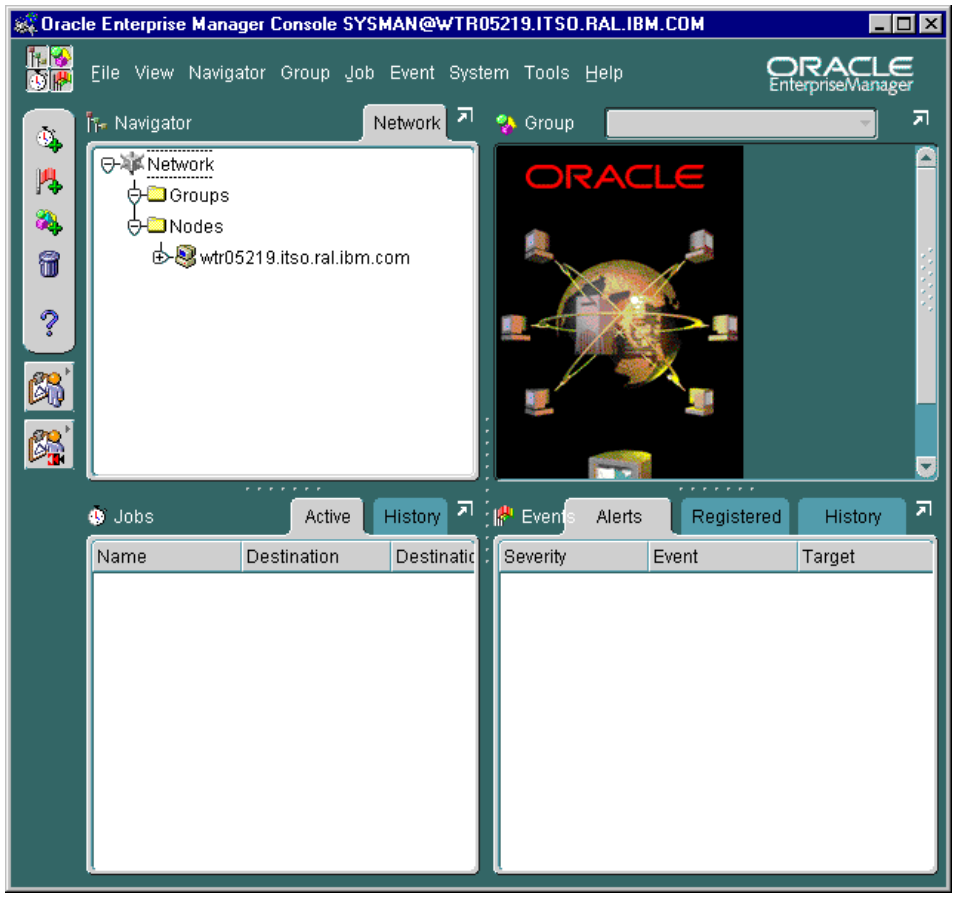


Figure 91. Oracle Enterprise Manager console

11.1.3 Second tier

The Oracle Management Server is the second tier or the middle tier in Figure 90 on page 145. The Oracle Management Server provides centralized intelligence and distribution control between the administration console and managed nodes. The Oracle Management Server processes all management tasks and sees to it that they are distributed to intelligent agents on the managed nodes (third tier). If multiple Oracle Management Servers are used, they can balance and share the workload to ensure performance.

11.1.3.1 Common services

The following services belong to the second tier and help you manage your environment:

Repository
A repository is a set of tables in an Oracle database and is used as a a back-end store by the Oracle Management Server. The repository is created when you install an Oracle Management Server in Oracle Enterprise Manager. The repository tables can be installed in any database accessible to the Oracle Management Server.

Job service
With job service you can create and manage jobs, schedule their execution and view and share information about defined jobs with other administrators connected to the repository. When a scheduled job is done (or failed) administrators will be notified through console alert, e-mail, or pager.

Event service
The event service monitors the Oracle environment for events. These events are, for example, loss of service, shortage of storage, high CPU usage, etc. You will be notified of an event by console alert, e-mail, or pager. It is possible to set up a "fixit" job to run automatically in response to an event.

Security service
The security service parameters in Oracle Enterprise Manager are defined for services, objects and administrators. All administrator accounts are defined by a super administrator who creates and defines the permissions of all the repository's administrators. A super administrator is able to access any object and control its security parameters, even though they are owned by other administrators.

Discovery service
Discovery service makes it possible for Oracle Enterprise Manager to discover all the databases and services running on the different managed nodes. Intelligent agents search for services running on nodes they manage and then communicate the information to the Oracle Management Server. The result of the search will be shown in a hierarchical tree in the Navigator window of the console. See Figure 91 on page 147.

11.1.4 Third tier

The third tier is the managed nodes. In this redbook the third tier is the Oracle Parallel Server (the Longhorn installation). The third tier communicates with Oracle Enterprise Manager using intelligent agents that reside on the managed nodes. The intelligent agent monitors the services in the node for events and executes jobs sent by the console via the Oracle Management Server.

11.2 Installing Oracle Enterprise Manager

You can install the Oracle Enterprise Manager client on a PC running Windows NT, Windows 95 or Windows 98 workstation. The Management Server must be installed on a PC running Windows NT and Oracle8*i* Server. When you install the Management Server you need to create the Repository Database. It is possible to install the Repository Database, the Oracle Management Server (OMS) and the Console on the same machine. We have made this choice.

The PC which will host the Repository Database, the Oracle Management Server (OMS) and the Console need to have at least a Pentium II 200 MHz, 128 MB of memory, and 1.5 GB of disk free.

Complete the installation process in 11.2.1, “Install of Oracle8i Enterprise Edition”

11.2.1 Install of Oracle8*i* Enterprise Edition

To handle a repository database, you have to install Oracle8*i* Server or Oracle8*i* Server Enterprise Edition. Follow the installation process from Chapter 7, “Installing Oracle8i Enterprise Edition” on page 61, you will not have any OPS specific feature. In the product components list, select and install:

- *Oracle8i* Server with the options: Database Configuration Assistant, and Oracle Intelligent Agent
- NET8 Client and Server
- Oracle Utilities
- Oracle Universal Installer

No product option is required.

11.2.2 Creating a database

The Oracle Enterprise Manager repository must reside in a database. Therefore, create a database using Oracle Database Configuration Assistant (as shown in 8.5, “Creating a database script using Oracle DB Configuration Assistant” on page 109).

Choose a Multipurpose database, with a Dedicated Server Mode. We have named it OEMrep. We have kept the defaults, the tablespaces information, and the SGA.

11.2.3 Network

Make sure that you can ping the nodes of the OPS cluster from the machine where you are going to install OEM (fill in the /etc/hosts as shown in 9.5, “Client deployment” on page 138).

Adapt the TNSNAMES.ORA with the OPS nodes services as shown in 9.5, “Client deployment” on page 138. Keep the service entry for the local Database (for example, OEMrep).

11.2.4 Installing Oracle Enterprise Manager

We will now install Oracle Enterprise Manager and create a repository:

1. Insert the Oracle Enterprise Manager Version 2.1 CD in the CD-ROM drive. The OEM installation window will appear; see Figure 92.

Figure 92. Installation window

2. If you are new to Oracle Enterprise Manager, we recommend that you click **Browse Documentation** then **Quick Tour**. Quick Tour will take you through Oracle Enterprise Manager and it is an excellent way to learn about the product quickly and easily.

If you are familiar with the product, click **Install/Deinstall Products**. You will see a window similar to Figure 93.

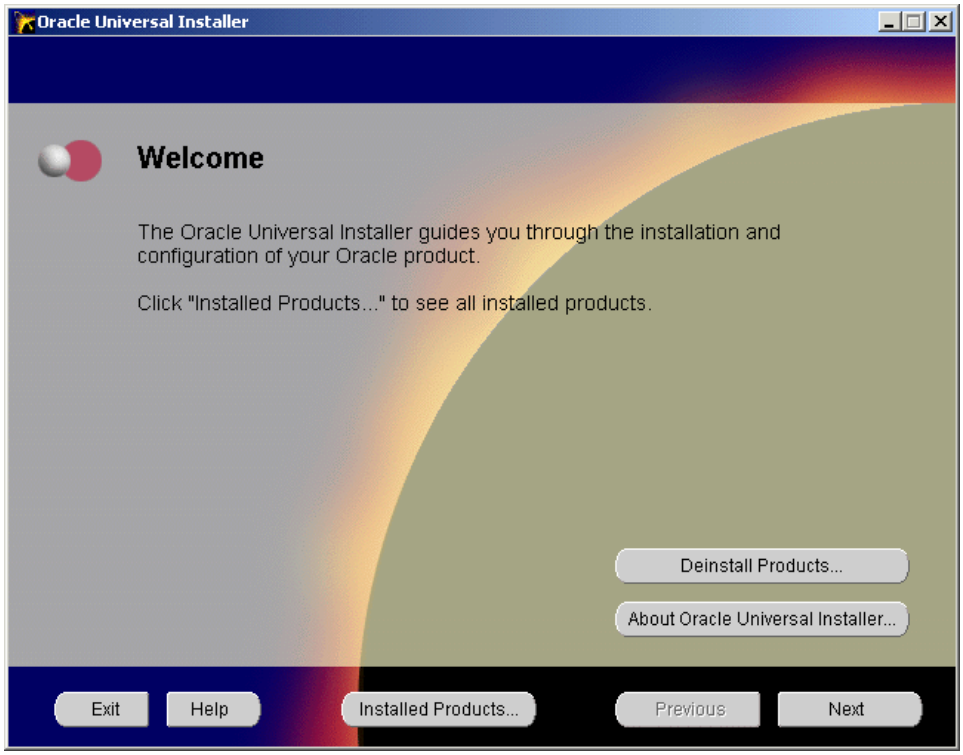


Figure 93. Welcome window

3. On the Welcome window, click **Next**. You will see a window similar to Figure 94.

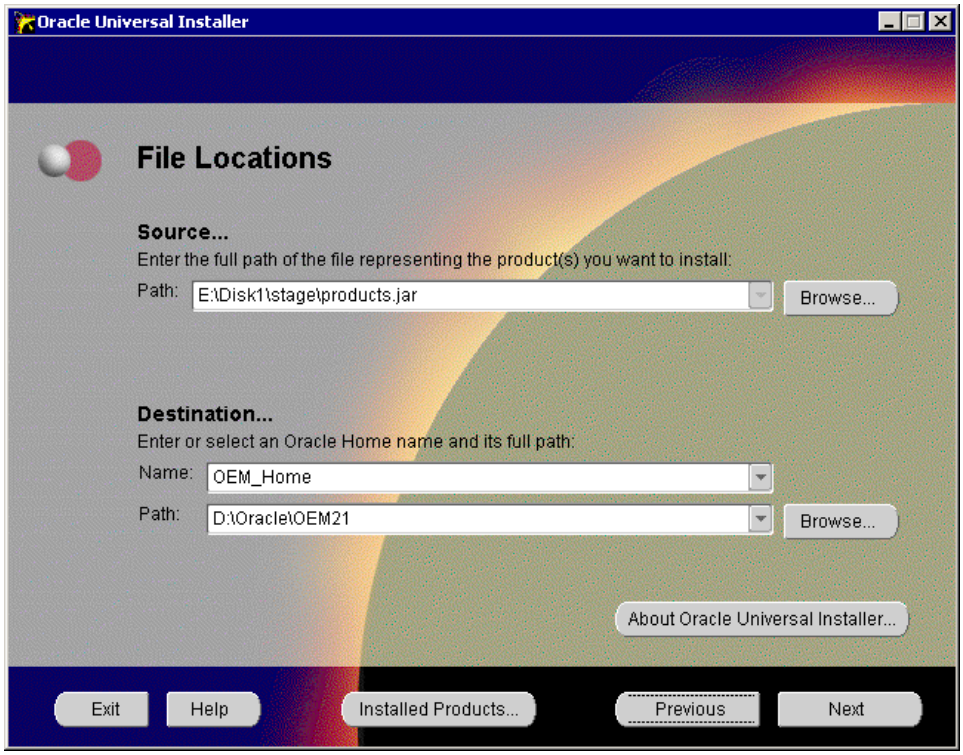


Figure 94. File Locations window

4. In the File Locations window, input the Oracle Home name for OEM and enter a path where you want to install OEM. For example, enter the name, `OEM_Home` for this path. The default path is your Oracle8*i* path, for our example, change it to `d:\Oracle\OEM21`. Click **Next**. You will see a window similar to Figure 95.

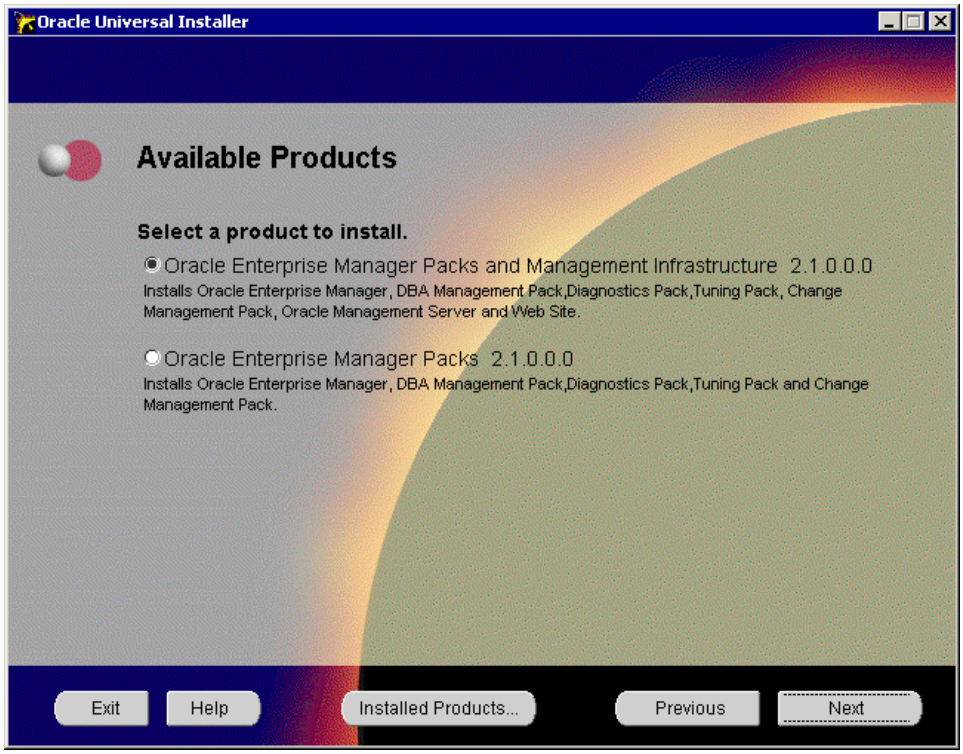


Figure 95. Available Products window

5. Select **Oracle Enterprise Manager Packs and Management Infrastructure 2.1.0.0.0**. We chose this selection because we have both the Management Server and the console on the same machine. Click **Next**. You will see a window similar to Figure 96 on page 154.

   Note: If you want these two functions on different machines, choose Enterprise Manager Pack to install the client only.

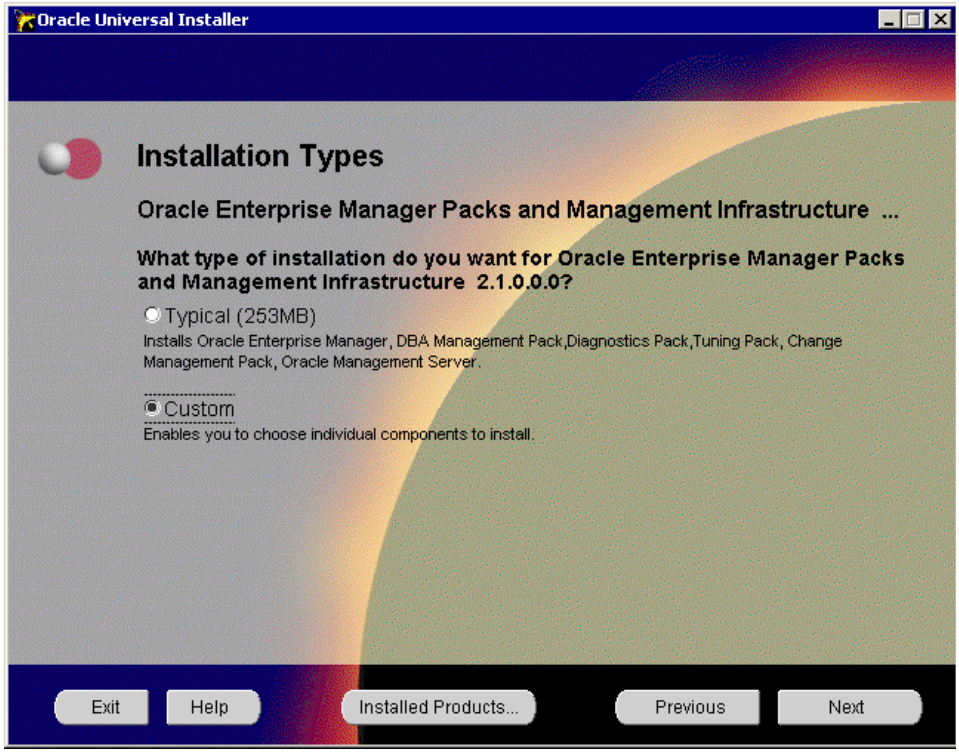


Figure 96. Installation Types window

6. Select **Custom** and click **Next** to continue. You will see a window similar to Figure 97.

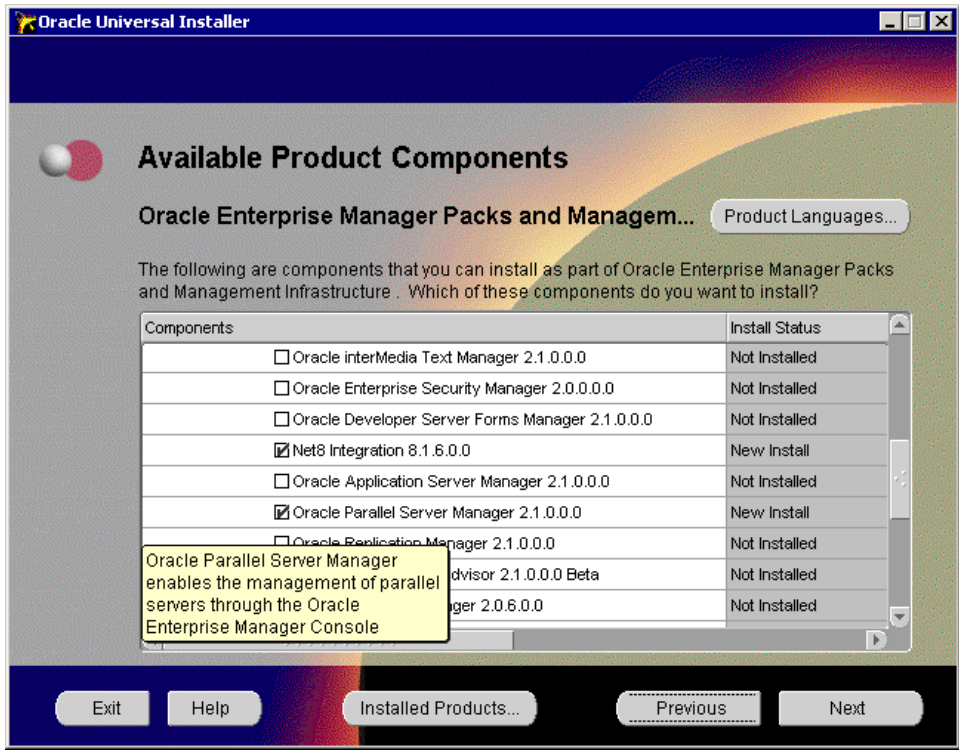


Figure 97. Product Components window

7. Select the following product components to install:
   - Oracle Management Server
   - Oracle Diagnostic Pack
   - Oracle Tuning Pack
   - Oracle Change Management Pack
   - Option Oracle Parallel Server Manager
   - Oracle DBA Management Pack

   Click **Next** to continue. You will see a window similar to Figure 98.

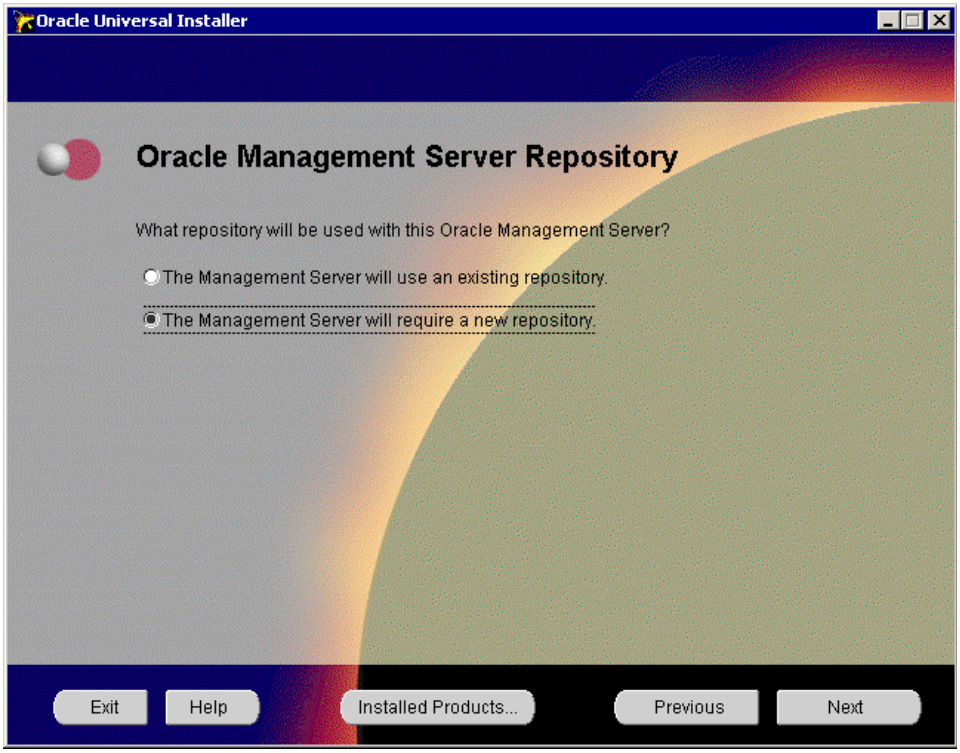


Figure 98. Oracle Management Server Repository window

8. Select **The Management Server will require a new repository** and click **Next**.
9. At the Installation Summary window, click **Install**. You will see a window similar to Figure 99.

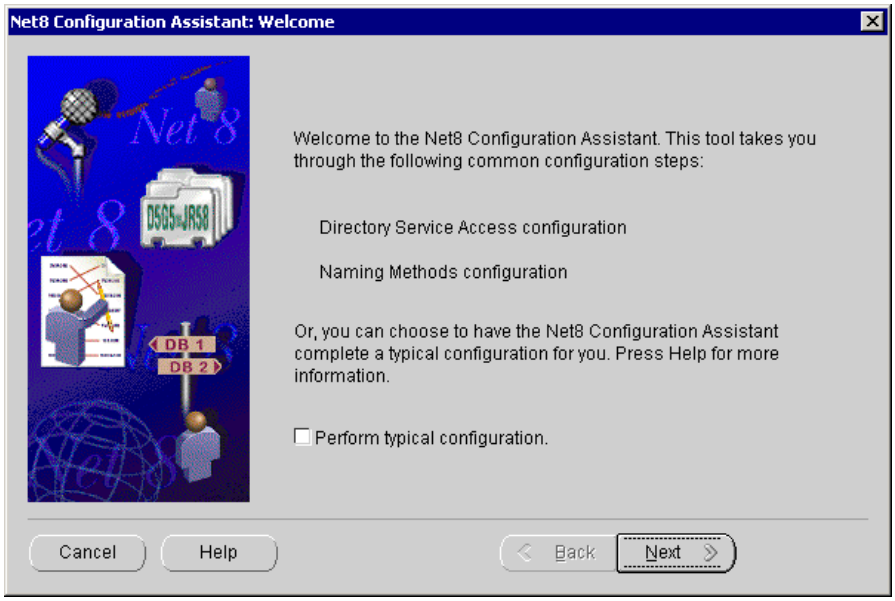


Figure 99. Net8 Configuration Assistant: Welcome window

10.Click **Next**. You will see a window similar to Figure 100.

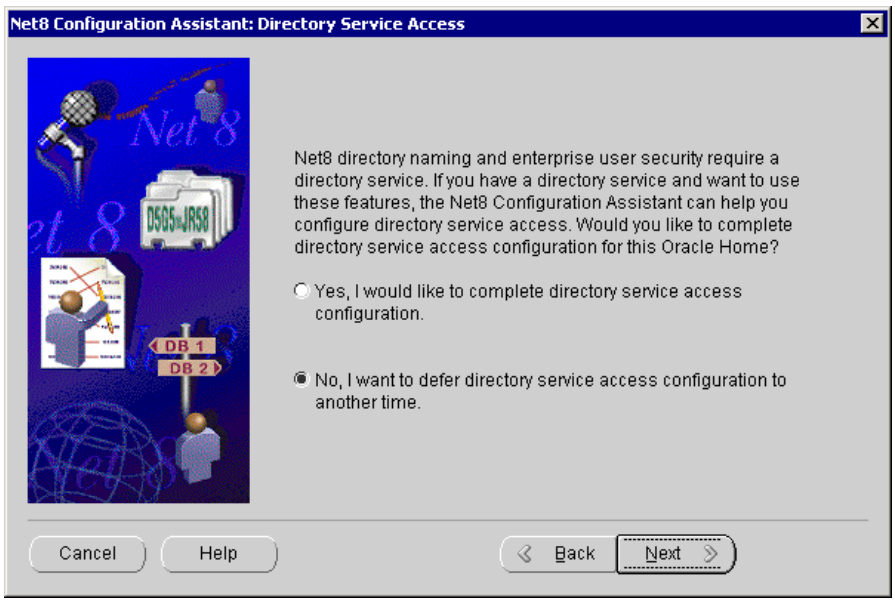


Figure 100. Net8 Configuration Assistant: Directory Service Access window

11. Select **No, I want to defer directory service access configuration to another time**. Click **Next** to continue. You will see a window similar to Figure 101.

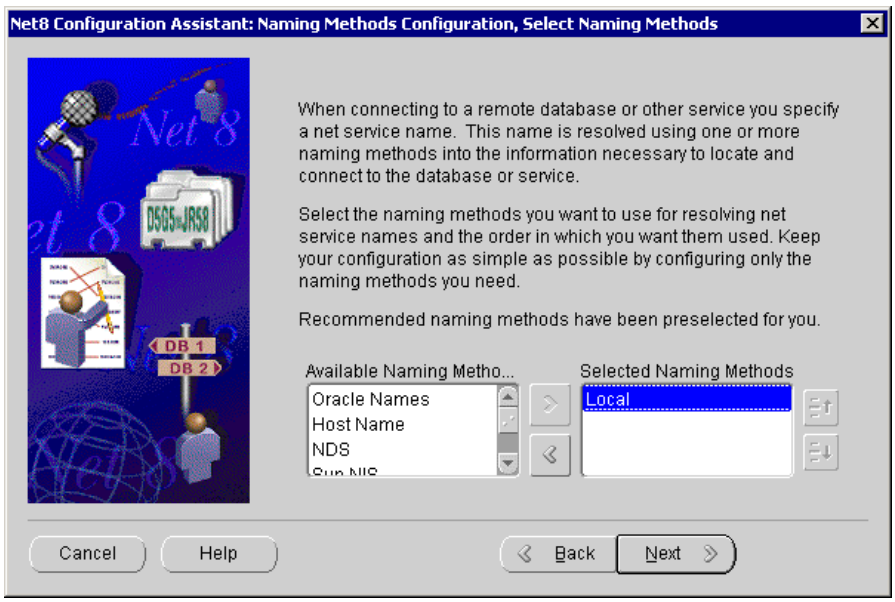


Figure 101. Naming Methods Configuration, Select Naming Methods window

12. Under Selected Naming Methods, only Local should be selected. Click **Next** to continue. You will see a window similar to Figure 102.

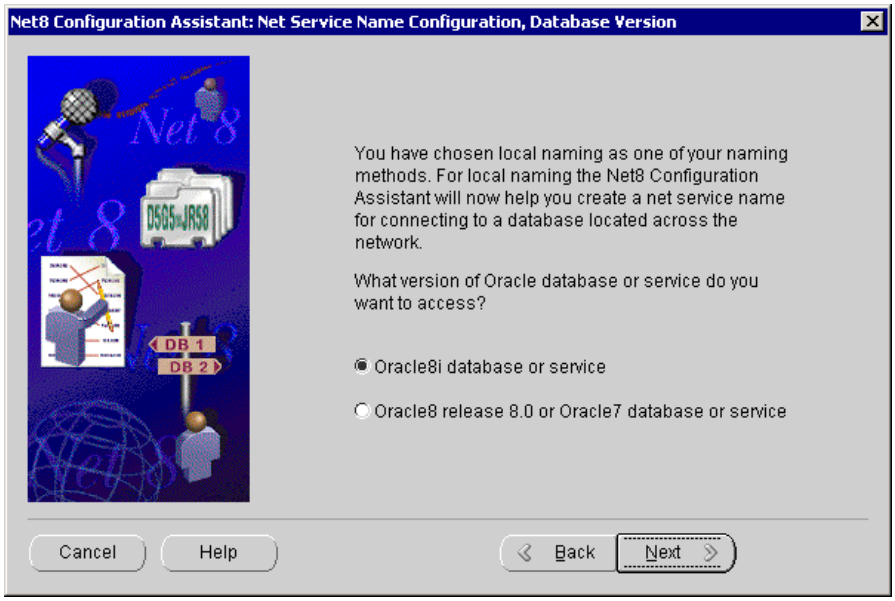


Figure 102. Net Service Name Configuration, Database Version window

13. At this window, you will select the version of Oracle database or service you intend to access for your repository. For our example, select **Oracle8*i* database or service**. Click **Next**. You will see a window similar to Figure 103.

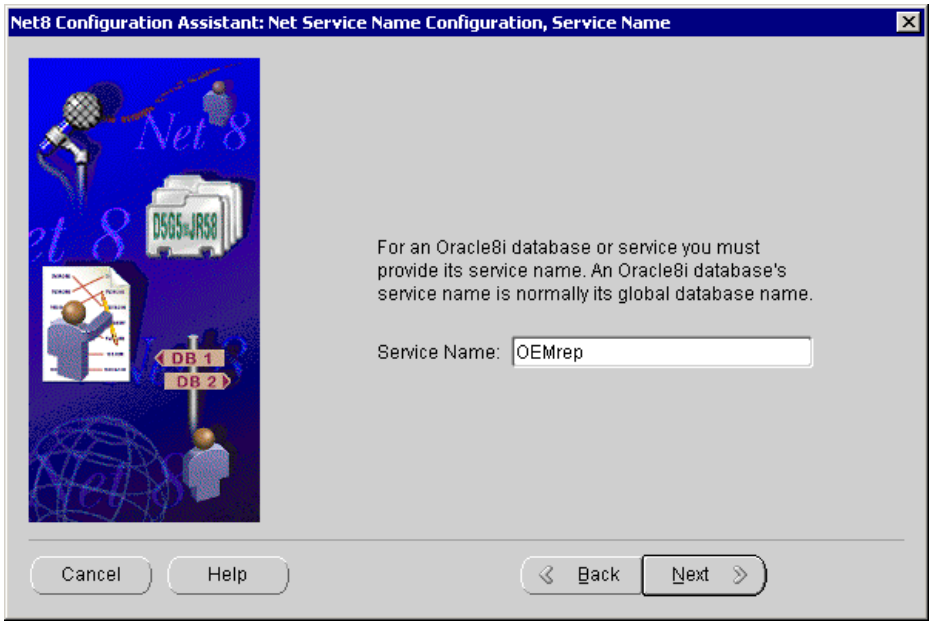


Figure 103. Net Service Name Configuration, Service Name window

14.Type `OEMrep` as the Service Name. Click **Next**.

15.Select **Protocols**. You will see a window similar to Figure 104.

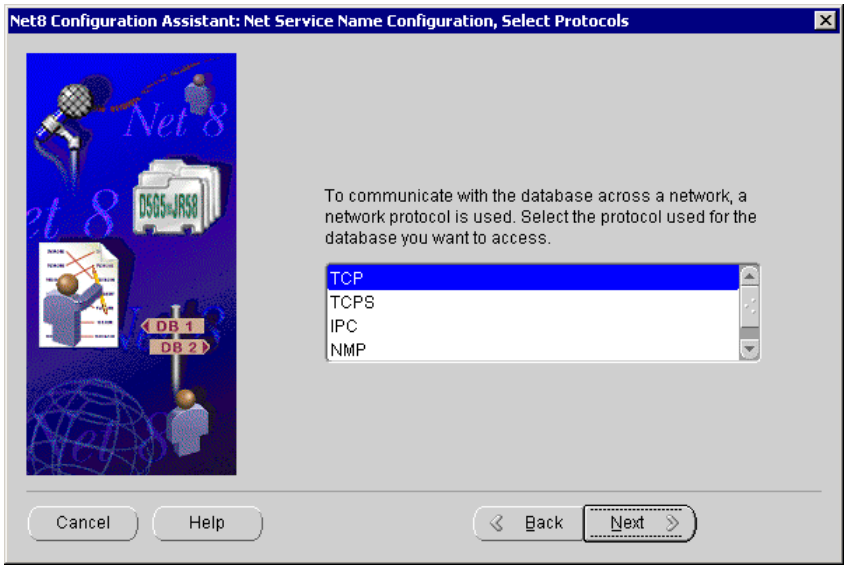


Figure 104. Net Service Name Configuration, Select Protocols window

16. Select **TCP** for the network protocol. Click **Next**. You will see a window similar to Figure 105.

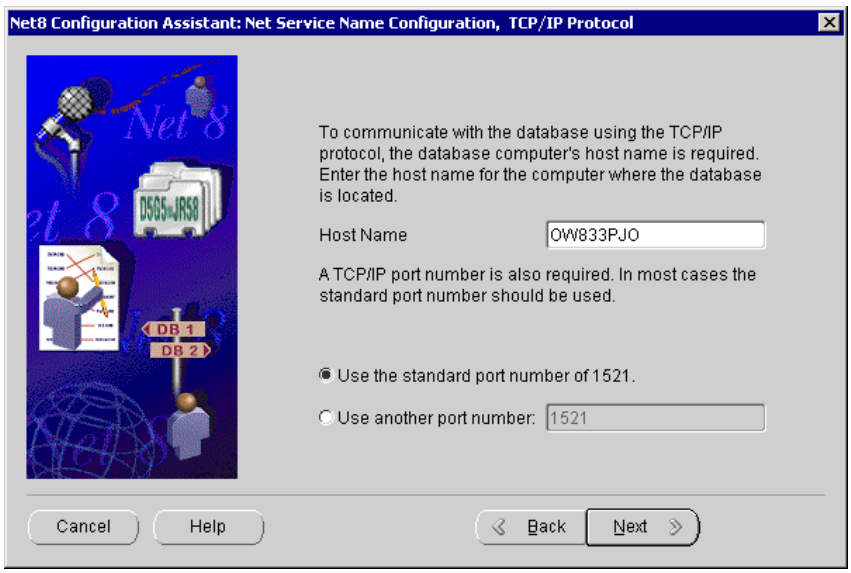


Figure 105. Net Service Name Configuration, TCP/IP Protocol window

17. In the Host Name field, type the host name of the computer where the database is located. In our example, type `OW833PJO`. Select **Use the standard port number of 1521**. Click **Next** to continue. You will see a window similar to Figure 106.

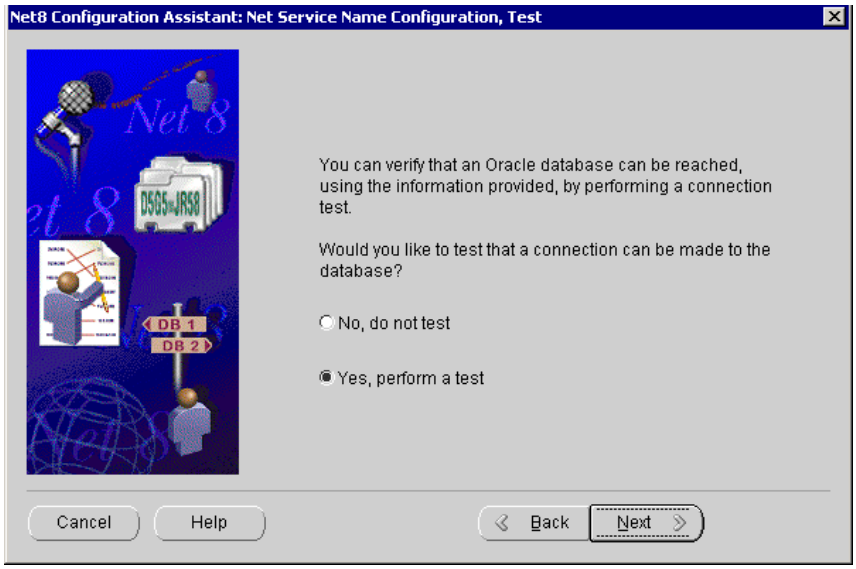


Figure 106. Net Service Name Configuration, Test window

18. Test the connection to the database. Select **Yes**, perform a test. If your test is successful, click **Next** to continue. You will see a window similar to Figure 107.

    Note: If your test is not successful, either you have made a mistake at your host or DB SID.

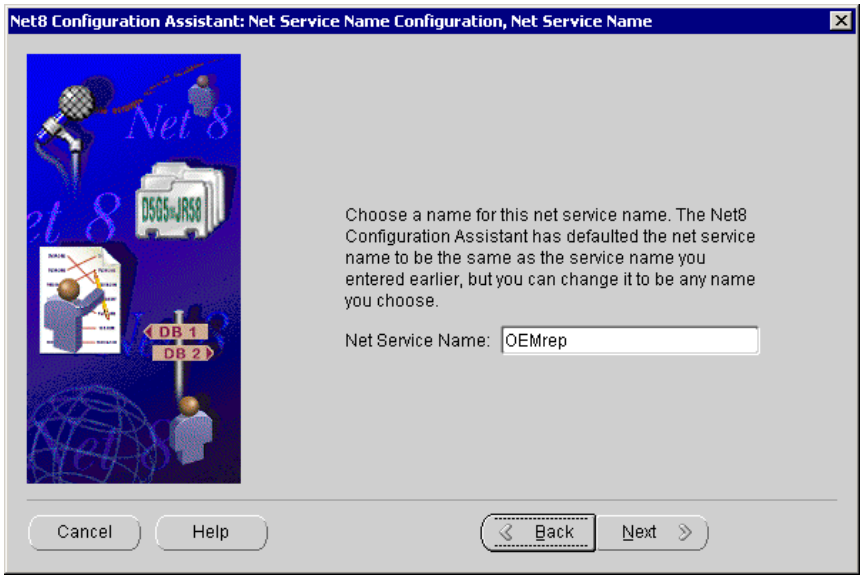


Figure 107. Net Service Name Configuration, Net Service Name window

19. In the Net Service Name field, type `OEMrep`. Click **Next**. You will see a window similar to Figure 108.

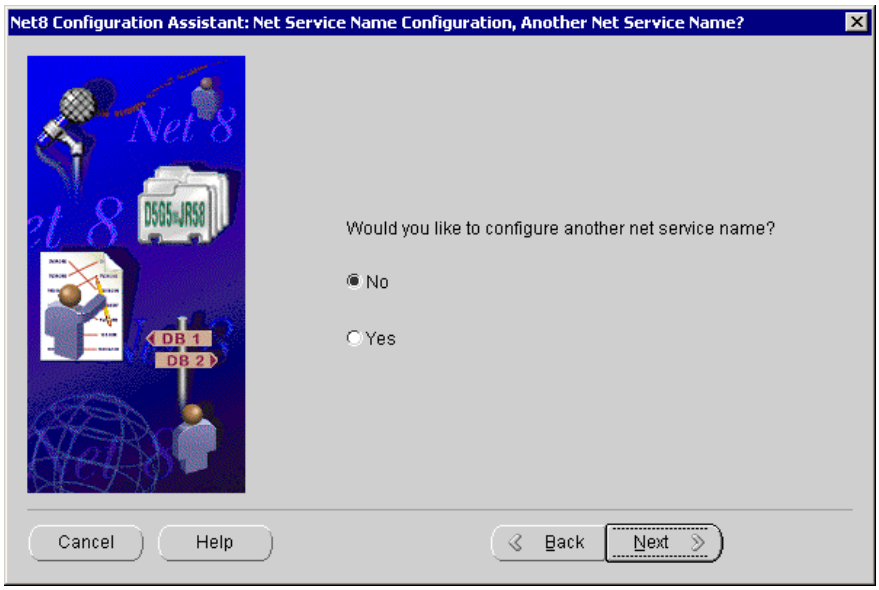


Figure 108. Net Service Name Configuration, Another Net Service Name window

20. Select **No** and click **Next** to continue.

21. Net8 Service Name Configuration completes. Click **Next**.

22. Naming Methods Configuration completes. Click **Next**.

23. Net8 Configuration completes. Click **Finish**. You will see a window similar to Figure 109.

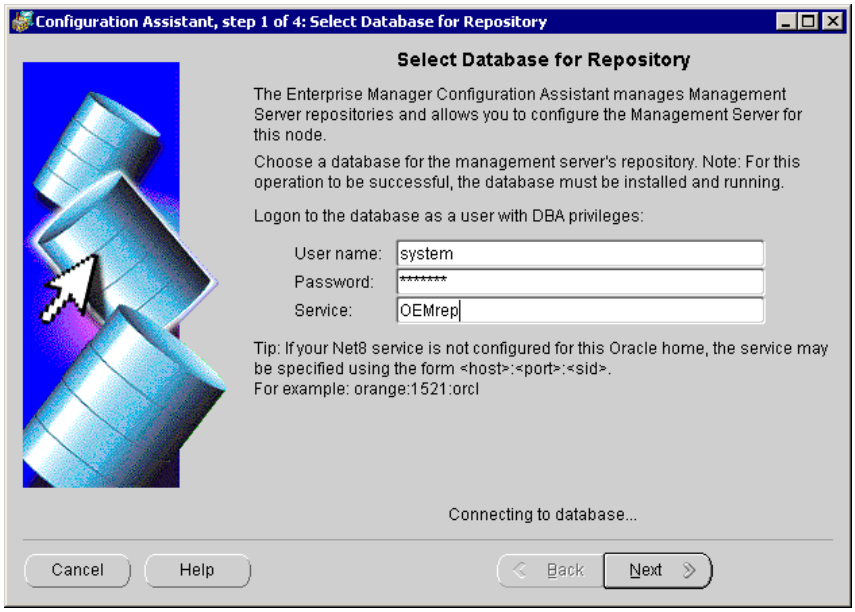


Figure 109. Select Database for Repository window

24. You must now choose the database you want to use for the repository. We will choose the database we just created. To access that database you will have to provide a user name and password with DBA (database administrator) privileges. For our example, the username is system, the password is manage, and service is OEMrep.

    Click **Next**. You will see a window similar to Figure 110.

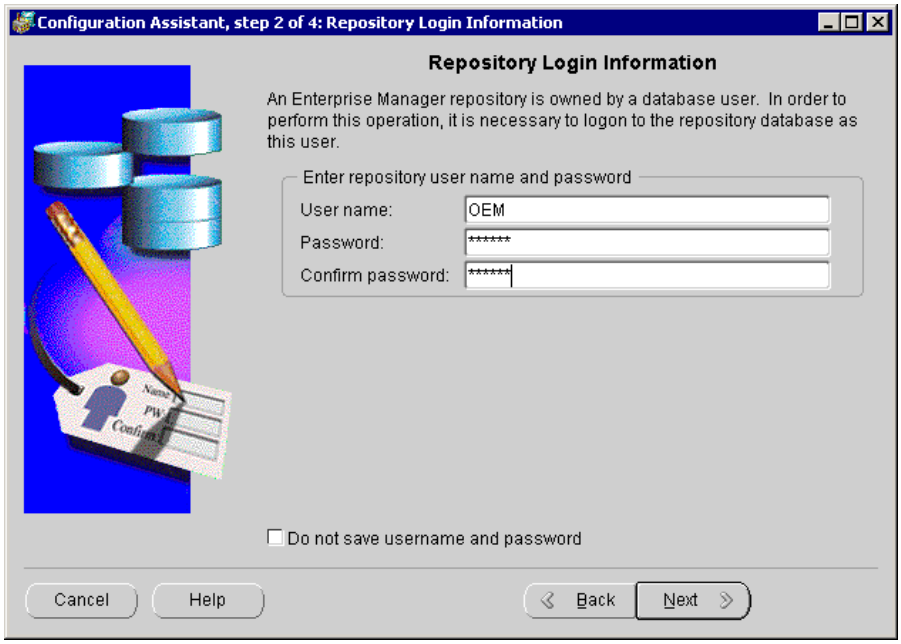


Figure 110. Repository Login Information window

25. The repository is owned by a database user. Enter the user name and password for the user you want to have as owner of the repository. For our example, type `OEM` for the username and `oracle` for the password. Type `oracle` again for the Confirm password. Click **Next** to continue. You will see a window similar to Figure 111.

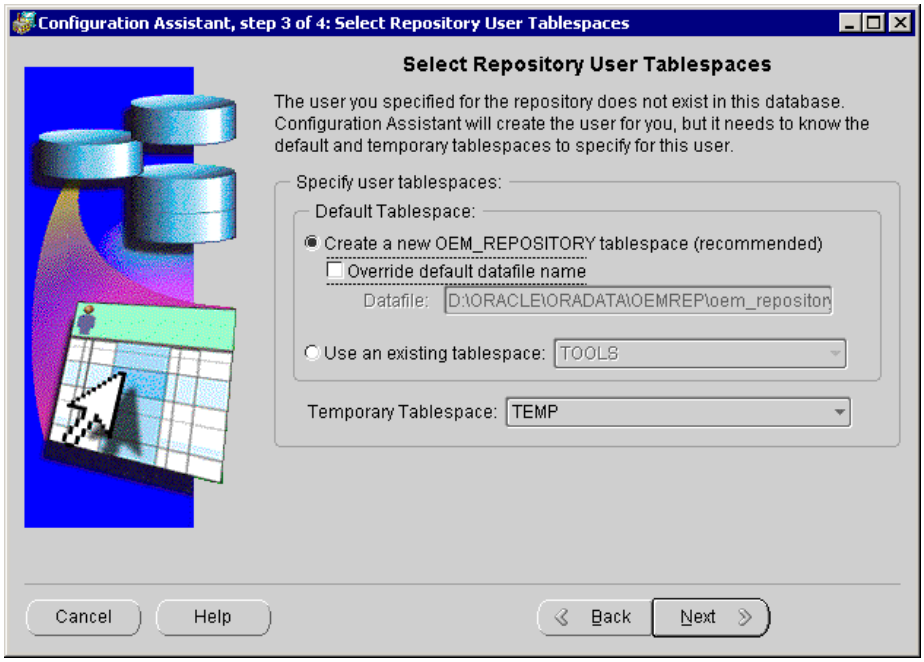


Figure 111. Select Repository User Tablespaces window

26. From the Select Repository User Tablespaces window, select **Create a new OEM_REPOSITORY tablespace (recommended)**. Change Temporary Tablespace to TEMP and click **Next** to continue. You will see a window similar to Figure 112.

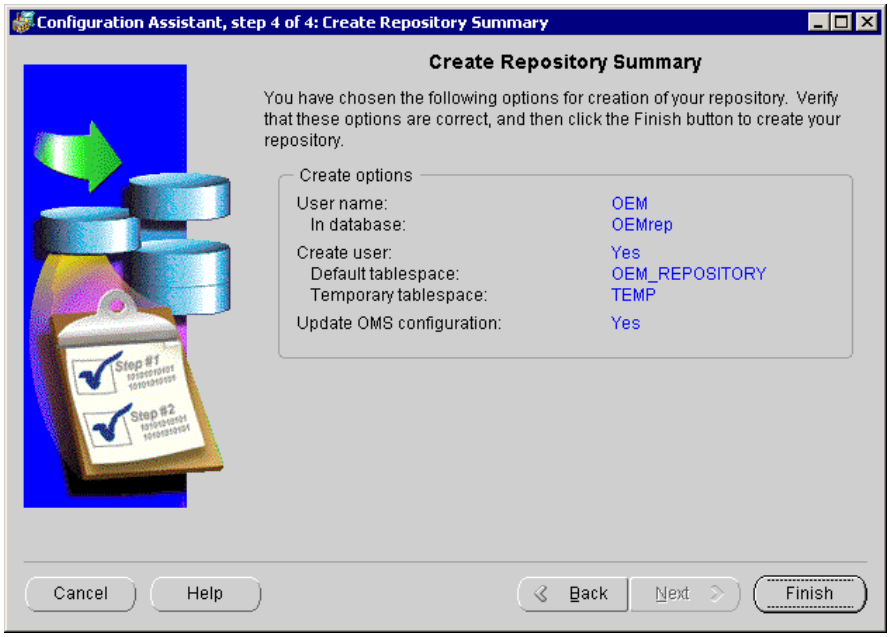


Figure 112. Create Repository Summary window

27. Review the options you chose in the Create Repository Summary and click **Finish**. A Configuration Assistant Progress window appears. Wait until the processing completes and then click **Close**. You will see a window similar to Figure 113.

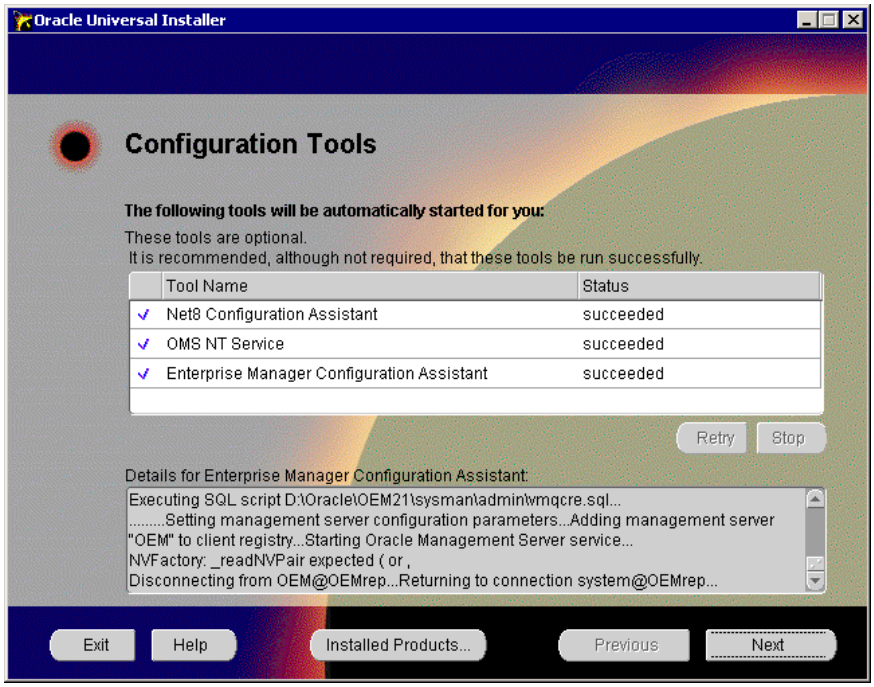


Figure 113. Configuration Tools window

28. At the Configuration Tools window, several tools will automatically start. Wait until these tools complete executing and that each item listed under Status indicates succeeded. Click **Next**. You will see a window similar to Figure 114.

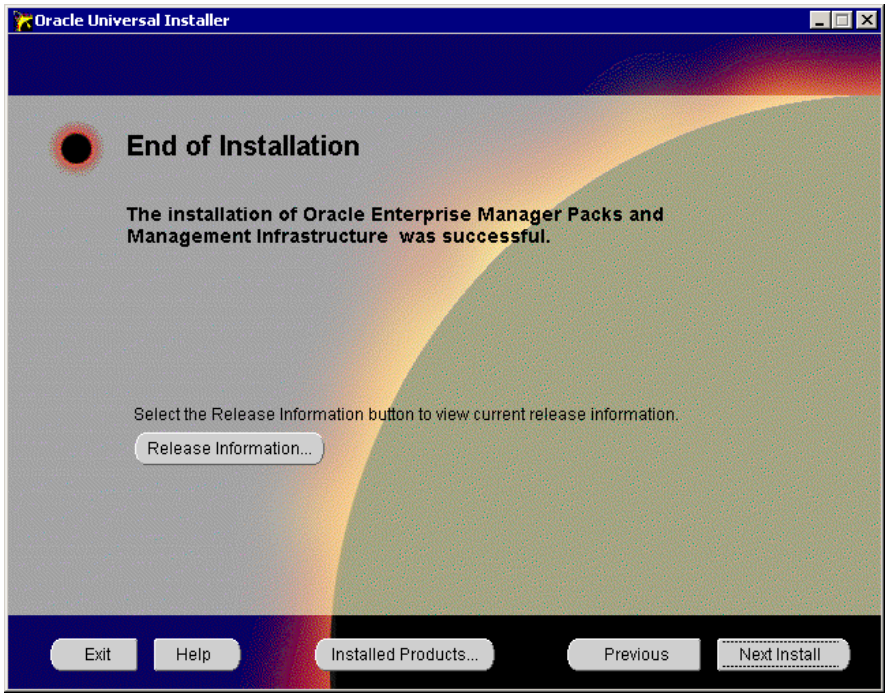


Figure 114. End of Installation window

29.Click **Exit**.

Appendix A. Hints and tips

This appendix will provide hints and tips for possible problems during and after installation.

A.1 Problem determination

The following may be of help during installation:

A.1.1 OSD installation

The OSD could not be installed on one or more nodes:

- If you are upgrading the OSD, remember to stop the OracleCMService and OracleNMService.
- Check for network connectivity between the nodes.

The OSD service could not be started on one or more nodes:

- Typical indication is a pop-up message box stating that the NM.dll was not loaded correctly.
- A common cause is that the heartbeat disk was not configured correctly.
- A shared-disk partition must be defined in *database*.tbl. (*database* should be the actual name of your database)

  Each of the shared disk partitions needs a name to distinguish it from other partitions. On your local disk, these names are simply the drive letters C:, D:, etc. If you have lots of shared disk partitions, however, you will run out of drive letters. So Windows (DOS actually) provides a way to assign a symbolic name to a specific disk partition. Oracle references the different partitions by name, not by physical location on the disk. This makes it easier to increase partition sizes, move them, etc. Setlinks.exe is a utility that was provided by Oracle with releases prior to 8.1.6. You would manually create a file called <database>.tbl which had the names and disk partition offsets. Setlinks would read that file and make the DOS calls to set up the mappings. Then when Oracle did I/O, it could use the symbolic names and Windows would map them to the correct partitions on the disks.
- The Oracle setlinks utility must have been run.
- A symbolic name must be entered on the osdconfig panel.

A.1.2 Oracle8*i* installation

Oracle Parallel Server is not an option from the Oracle Universal Installer:

- OSD was not installed correctly.
- OracleCMService and OracleNMService are not started.

A.1.3 Quick recovery

- Stop database, Oracle services and OSD services.
- Restart those services and the database.
- If a problem occurs at a lower level, you may need to restart that level and all levels above it.

A.1.4 Isolation to layer

- Errors in OSD log files?
- Which services are up? Which are down?
- Interconnect error will show up in the CM.log.
- Shared DASD - typically only affects Oracle services but may also affect NMSRVR if CM heartbeat disk partition is affected.

A.2 General procedures

The following will describe briefly how to carry out different procedures:

A.2.1 Effects of stopping resources

Stopping a node:

- Database is still accessible but with possible performance degradations.

Stopping shared DASD:

- Database is not accessible but services will still be running.

Interconnect hub:

- Database is accessible only from the first node. Databases on the other nodes will terminate.

Network cable:

- Database is not accessible from that node, but the services will still be running.

Shared DASD cable:

- Database is not accessible from that node, but services will still be running.

A.3 Miscellaneous tips

The following will provide miscellaneous tips about Oracle installation, determining version numbers, creating databases and installing additional products.

A.3.1 Version numbers

To determine the version numbers of Windows NT click **Start** -> **Control Panel** -> **Help** -> **About Windows NT.**

To determine the version of the OSD, see OSDInstall Dir\version.txt. It should say IBM Netfinity Advanced Cluster Enabler for OPS - Version 2.1.

To determine the version of Oracle click **Start** -> **Programs** -> **Oracle-OraHome81** - > **Release Notes.**

A.3.2 Creating databases

When creating a database with Oracle database Configuration Assistant, the SID field is the sid prefix. Do not add the instance number. If the database is called OP (database_name = OP) then the SID prefix field should also be OP.

A.3.2.1 Deleting a database

You can delete a database in two ways. You can use the Oracle Database Configuration Manager or you can do it manually. To do it manually carry out the following steps:

- Delete the directory at c:\oracle\admin\db-name
- Delete files from c:\dbcreate\createdb1\db-name*.*
- Delete files from c:\oracle\ora81\assistants\dbca\db-name*.*

A.3.3 Installing additional products

You install additional products using Oracle Universal Installer. It is important, when installing additional products, not to deselect already installed products. If you deselect already installed products, they will be uninstalled.

Appendix B. Special notices

This publication is intended to help customers, business partners and IBM employees who will be planning or installing Oracle Parallel Server, Oracle8*i* Enterprise Edition 8.1.6, IBM Netfinity Advanced Cluster Enabler for OPS V2.1 and Microsoft Windows 2000 Advanced Servers on IBM Netfinity servers. The information in this publication is not intended as the specification of any programming interfaces that are provided by these products. See the PUBLICATIONS section of the IBM Programming Announcement for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been

reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

IBM	e(logo) @
Redbooks	Redbooks Logo 
RS/6000	Netfinity
SP	ServeRAID

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere.,The Power To Manage., Anything. Anywhere.,TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

Oracle, Oracle Parallel Server, and Oracle8*i* are registered trademarks of Oracle Corporation.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, Windows 2000 and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

Other company, product, and service names may be trademarks or service marks of others.

Appendix C. Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

C.1 IBM Redbooks

For information on ordering these publications see “How to get IBM Redbooks” on page 181.

- *Implementing Oracle Parallel Server on Netfinity Servers*, SG24-5449
- *Oracle8i Parallel Server on IBM SP Systems: Implementation Guide*, SG24-5591
- *Tuning Netfinity Servers for Performance - Getting the most out of Windows 2000 and Windows NT 4.0*, SG24-5287
- *Netfinity Server Disk Subsystems*, SG24-2098

C.2 IBM Redbooks collections

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at ibm.com/redbooks for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	Collection Kit Number
IBM System/390 Redbooks Collection	SK2T-2177
IBM Networking Redbooks Collection	SK2T-6022
IBM Transaction Processing and Data Management Redbooks Collection	SK2T-8038
IBM Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
IBM AS/400 Redbooks Collection	SK2T-2849
IBM Netfinity Hardware and Software Redbooks Collection	SK2T-8046
IBM RS/6000 Redbooks Collection	SK2T-8043
IBM Application Development Redbooks Collection	SK2T-8037
IBM Enterprise Storage and Systems Management Solutions	SK3T-3694

C.3 Referenced Web sites

These Web sites are also relevant as further information sources:

- http://www.pc.ibm.com/us/netfinity/parallel_server.html
- http://www.ibm.com/pc/us/products/server/download.html
- http://www.pc.ibm.com/us/netfinity

- http://www.pc.ibm.com/support
- http://www.pc.ibm.com/us/files
- http://www.pc.ibm.com/us/netfinity/storesvr.html#nrs
- http://www.pc.ibm.com/ww/netfinity/fibrechannel/
- http://www.pc.ibm.com/us/eserver/xseries/

How to get IBM Redbooks

This section explains how both customers and IBM employees can find out about IBM Redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** ibm.com/redbooks

  Search for, view, download, or order hardcopy/CD-ROM Redbooks from the Redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

  Redpieces are Redbooks in progress; not all Redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

  Send orders by e-mail including information from the IBM Redbooks fax order form to:

	e-mail address
In United States or Canada	pubscan@us.ibm.com
Outside North America	Contact information is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	Country coordinator phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Fax Orders**

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	Fax phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and Redbooks by accessing the IBM Intranet Web site at http://w3.itso.ibm.com/ and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at http://w3.ibm.com/ for redbook, residency, and workshop announcements.

IBM Redbooks fax order form

Please send me the following:

Title	Order Number	Quantity

First name	Last name	
Company		
Address		
City	Postal code	Country
Telephone number	Telefax number	VAT number

☐ Invoice to customer number _____

☐ Credit card number _____

Credit card expiration date	Card issued to	Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Index

P

R

S

T

V

W

X

IBM Redbooks review

Your feedback is valued by the Redbook authors. In particular we are interested in situations where a Redbook "made the difference" in a task or problem you encountered. Using one of the following methods, **please review the Redbook, addressing value, subject matter, structure, depth and quality as appropriate.**

- Use the online **Contact us** review redbook form found at **ibm.com**/redbooks
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Document Number **Redbook Title**	SG24-5449-01 Oracle Parallel Server and Windows 2000 Advanced Server on IBM Netfinity
Review	
What other subjects would you like to see IBM Redbooks address?	
Please rate your overall satisfaction:	O Very Good O Good O Average O Poor
Please identify yourself as belonging to one of the following groups:	O Customer O Business Partner O Solution Developer O IBM, Lotus or Tivoli Employee O None of the above
Your email address: The data you provide here may be used to provide you with information from IBM or our business partners about our products, services or activities.	O Please do not use the information collected here for future marketing or promotional contacts or other communications beyond the scope of this transaction.
Questions about IBM's privacy policy?	The following link explains how we protect your personal information. **ibm.com**/privacy/yourprivacy/

Oracle Parallel Server and Windows 2000 Advanced Server on IBM Netfinity

(0.2"spine)
0.17"<->0.473"
90<->249 pages

Oracle Parallel Server and Windows 2000 Advanced Server on IBM Netfinity

Set up Oracle Parallel Server for Windows 2000 Advanced Server

Set up and configure Netfinity Fibre Channel Storage Manager

Configure load balancing and failover

This redbook provides an overview of the recent version of Oracle8i Parallel Server (OPS) for Windows 2000 running on IBM Netfinity.

First you will be given an introduction to IBM's Oracle Parallel Server solution where we explain the solution provided by IBM. Next, we provide an understanding of the Oracle and Oracle Parallel Server Architecture. Then we discuss the planning phase, the hardware and software requirements, and most important, how to configure your hardware and software.

Once the hardware components and the operating system are installed, we discuss OPS software installation in detail, including the installation of the Operating System Dependent (OSD) layer, and the installation of Oracle8i and Oracle Enterprise Manager. In order for you to test and verify the configuration, we also create a database and test it in parallel mode.

This redbook includes the configuration of OPS in a Netfinity Fibre Channel High Availability environment, the installation and configuration of Netfinity Fibre Channel Storage Manager, and the setup and configuration of Load Balancing and Failover.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-5449-01 ISBN 0738419060