

ORACLE 10G AND LINUX NFS ON DELL™ NX4

Solution Guide for Oracle
RAC 10g and Linux NFS on
Dell NX4

Dell Inc.

Visit dell.com/NX4 for more information and additional resources

Copyright © 2008 Dell Inc. THIS WHITE PAPER IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND.



Contents

	About this Document	9
Chapter 1	Solution Overview.....	10
	Technology solution.....	11
	Solution advantages.....	12
	Terminology	13
Chapter 2	Solution Reference Architecture	15
	Overall architecture	16
	General characteristics	16
	Network architecture.....	18
	Virtual local area networks.....	18
	Switches	18
	Dell NX4 Data Mover ports.....	18
	Storage architecture.....	19
	High availability and failover	19
	RAID type and RAID group configuration.....	20
	Disk volume setup	23
	File systems, exports, and mount points	23
	Database server architecture.....	24
	Oracle RAC 10g server network architecture	24
	High availability and failover	24
	Database software.....	25
	High availability and failover.....	26
	Storage layer	26
	Network layer	26
	Host layer.....	26
	Site level protection	26
	Hardware and software resources.....	27
	Hardware resources	27
	Software resources.....	27
Chapter 3	Solution Best Practices.....	29
	Solution architecture best practices	30
	Solution validation and performance testing overview.....	30
	Storage setup and configuration	33
	General disk drive recommendations.....	33
	Database file storage.....	33
	Volume management	33
	Data Mover parameter setup.....	34
	NFS mount point parameters	35
	Load distribution.....	35
	High availability	36
	Network setup and configuration	36

	Gigabit connection.....	36
	Virtual local area networks	37
	Network port configuration	37
	Network security.....	37
	Jumbo frames.....	37
	Database server setup and configuration	37
	Server BIOS.....	37
	Hyperthreading	37
	Memory	38
	Linux setup and configuration.....	38
	Virtual memory.....	39
	Oracle memory structure	39
	Static IP address.....	39
	Oracle database setup and configuration	39
	Oracle database file placement	39
	Oracle database initialization parameters	40
	Recommendation for Oracle database files	41
	Backup, recovery, and protect setup and configuration	41
	Virtual test/dev environment configuration.....	42
Chapter 4	Solution Applied Technologies	43
	Solution applied technologies	43
	Physical backup and recovery using Oracle Recovery Manager (RMAN)	45
	Full backup	45
	Full restore and recovery	46
	Incremental backup.....	46
	Incremental restore and recovery.....	46
	Logical backup and recovery using Celerra SnapSure	47
	Logical backup procedure.....	47
	Logical recovery procedure	47
	Advanced protect and recovery using Celerra Replicator (V2).....	48
	Setting up communication between NX4 Network Servers at production and remote sites	48
	Setting up communication between Data Movers at the production and remote sites.....	48
	Creating file system replication sessions between the production and remote sites	49
	Fail over to the remote site	49
	Recovering the database at the remote site	50
	Restart file system replication sessions from the remote to production site	51
	Fail back to the production site.....	51
	Virtual test/dev solution using Celerra SnapSure.....	52
Chapter 5	Conclusion.....	55
	Conclusion.....	56
Appendix A	Sample ks.cfg	57
	Sample ks.cfg	58

Figures

Figure 1	Oracle RAC 10g for Linux on NFS - system architecture	17
Figure 2	Dell NX4 Data Mover ports and traffic types	18
Figure 3	Oracle RAC 10g – traditional configuration	20
Figure 4	Oracle RAC 10g – one building block configuration	21
Figure 5	Oracle RAC 10g – two building blocks configuration	22
Figure 6	TPC-C user load scaling	31
Figure 7	CPU utilization chart	31
Figure 8	RAC scaling	32
Figure 9	Tested configurations – TPS	32

Tables

Table 1 Solution advantages.....	12
Table 2 Oracle database 10g solution terminology	13
Table 3 File system layout.....	23
Table 4 Database server network interface configuration	24
Table 5 Database file types and locations.....	25
Table 6 Hardware specifications	27
Table 7 Software specifications.....	27
Table 8 Database file storage recommendations	33
Table 9 NFS mount point options on database servers.....	35
Table 10 Oracle RAC 10g solution VLANs	37
Table 11 Database parameter initialization options	40
Table 12 Parameter and descriptions	43

About this Document

This solution guide provides an overview of the architecture, best practices, and implementation strategies for Oracle RAC 10g for Linux stored on Dell NX4 arrays over NFS developed by the Dell NAS Product Validation group.

Purpose

This guide provides an overview of an Oracle RAC 10g implementation for Linux using Dell™ NX4 over NFS as the back-end storage. Information in this document can be used as the basis for a solution build, white paper, best practices document, or training.

Audience

This document is intended for personnel, partners, and customers looking for a cost-effective way to implement production Oracle databases.

Scope

This document describes the architecture, best practices, and implementation strategies for Oracle RAC 10g for Linux stored on Dell NX4. Implementation instructions and sizing guidelines are beyond the scope of this document.

Related documents

The following documents, located on dell.com/NX4, provide additional, relevant information.

- ◆ *Advanced File Sharing and Management with the Dell NX4 – Whitepaper*
- ◆ *Email, Database and File Sharing on Dell NX4 – Reference Architecture*

The following third-party documents are available from Oracle at <http://www.oracle.com>:

- ◆ *Oracle Database Installation Guide 10g Release 2 (10.2) for Linux (x86-64)*
- ◆ *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide 10g Release 2 (10.2) for Linux*
- ◆ *Oracle RAC, Overview of Real Application Clustering Features and Functionality*
- ◆ *Database Performance with Oracle Database 10g Release 2*
- ◆ *Oracle Database Backup and Recovery Advanced User's Guide 10g Release (10.2)*

Information about VMware Infrastructure 3 is available at <http://www.vmware.com>.

Chapter 2 Solution Overview

This chapter presents these topics:

Technology solution 11

Technology solution

The Oracle RAC 10g for Linux over NFS solution is deployed on a Dell NX4 multi-protocol storage system. The storage access method is NFS. The NFS file systems on the NX4 are presented to the database servers. This enables a midsize enterprise to deploy the NX4 network-attached storage (NAS) architecture with NFS connectivity for its Oracle Database 10g RAC applications with lower cost and complexity than direct-attached storage (DAS) or a storage area network (SAN).

The Oracle Database RAC 10g on NX4 NFS solution is comprised of six individual solution components: consolidation, basic backup, advanced backup, advanced protect, resiliency, and test/dev environment.

The consolidation component details how the NFS storage is provisioned and presented to the database servers. The performance of the consolidation component is tested by using an industry-standard OLTP database performance benchmark, while providing real-world tuning on a reasonably priced and configured platform.

The basic backup component verifies the basic backup and restore capabilities using Oracle Recovery Manager (RMAN), as well as the resulting restore and recovery performance characteristics. This backup uses only the functionality provided by the database server and operating system software to perform backup and recovery. It uses the database server's CPU, memory, and I/O channels for all backup, restore, and recovery operations.

The advanced backup component verifies the backup and restore capabilities using Celerra SnapSure™, as well as the resulting restore and recovery performance characteristics. This component uses additional software components at the storage layer to offload the database server's CPU, memory, and I/O channels for all backup, restore, and recovery operations. It provides nearly instantaneous backup and restore operations.

The advanced protect component verifies the remote replication and recover capabilities using Celerra Replicator™ (V2), as well as the resulting performance characteristics. It uses additional software components at the storage layer to offload the database server's CPU, memory, and I/O channels for all replication and recovery operations. Because of the asynchronous nature of replication, full recovery of data in case of disaster is not assured.

The resiliency component tests the availability of the Oracle 10g RAC database in various failure scenarios, and it measures the resulting impact.

The test/dev component facilitates the creation of a copy of the Oracle 10g RAC database using the writeable snapshots feature of Celerra SnapSure. This component also verifies the impact of the test/dev database on the primary database.

Note: According to Oracle's *Support Position for Oracle Products Running on VMware Virtualized Environments*, Oracle has not certified any of its products on VMware virtualized environments. Oracle support will assist customers running Oracle products on VMware in the following manner: Oracle will provide support for issues that either are known to occur on the native OS, or can be demonstrated not to occur as a result of running on VMware.

Solution advantages

The NX4 Oracle RAC 10g for Linux solution offers the following advantages:

Table 1 Solution advantages

Benefit	Details
Lower total cost of ownership (TCO)	Reduces acquisition, administration, and maintenance costs more than comparable DAS or SAN
Greater manageability	Eases implementation, provisioning, and volume management
Simplified Real Application Clusters (RAC) implementation	Provides NFS-mounted shared file systems
High availability	Implements a clustering architecture that provides very high levels of data availability
Increased flexibility	Makes databases, or copies of database, available (via remounts) to other servers
Improved protection	Integrates availability and backup
Multi-protocol consolidation	Offers ability to use FC, iSCSI, NFS, and CIFS all within a single storage platform

Terminology

As you use the solution guide, it is important to understand certain key terms related to the components of the solution. [Table 2](#) provides definitions of the terms used in this document.

In addition, you can refer to respective vendor document for more information.

Table 2 Oracle database 10g solution terminology

Term	Definition
Building block (BB)	Building block specifies the number of physical disk drives on the Dell NX4 network server, the multiples of which can be configured and provisioned to the Oracle database servers.
Full backup	A non-incremental RMAN backup, which backs up all the data blocks in data files, irrespective of whether they are modified.
Incremental backup	A backup in which only modified blocks are backed up.
Link aggregation	A high availability feature based on the IEEE 802.3ad Link Aggregation Control Protocol (LACP) standard that allows Ethernet ports with similar characteristics to connect to the same switch to combine into single virtual device with a single MAC address.
Online Transaction Processing (OLTP)	The real-time high performing relational database system supporting thousands of concurrent users performing common set of transactions.
Oracle Recovery Manager (RMAN)	The Oracle-preferred method for Oracle database backup and recovery. It provides block-level corruption detection during backup and restore and uses file multiplexing to optimize performance and space consumption during backup.
Point-in-time recovery	The incomplete recovery of database files to a non-current time.
Physical storage backup	A full and complete copy of the database to different physical media.
SnapSure	A feature on the NX4 Network Server that provides read-only point-in-time copies of a file system.
Virtual Machine (VM)	A virtualized x86 PC on which a guest operating system and an associated application run. A VM is also a set of discrete files that primarily include a .vmx configuration file and one or many .vmdk virtual disk files.
VLAN	Logical networks that function independently of the physical network configuration.

Chapter 3 Solution Reference Architecture

This chapter presents these topics:

Overall architecture	16
General characteristics	16
Network architecture	18
Storage architecture.....	19
Database server architecture.....	24
High availability and failover.....	26
Hardware and software resources.....	27

Overall architecture

The Oracle RAC 10g solution uses the NFS protocol to access remote file systems on the Dell NX4. A dual-processor Intel server with 12 GB RAM can be used as the Oracle database server hosting the Red Hat Enterprise Linux operating system (OS), Oracle 10g RAC software, and the Oracle database software. The testing and development database server can be hosted on VMware ESX Server running Red Hat Enterprise Linux as the guest OS. An Dell NX4 is used as the storage device to store Oracle data files, redo log files, archived log files, flash recovery files, and cluster configuration files. All the shared volumes required by Oracle RAC are placed on Dell NX4 file systems accessed over NFS. The database servers access the files as if they are available on a local file system, but in reality the data transfer occurs over Ethernet using the TCP/IP communication protocol.

The Oracle RAC 10g solution consists of two sites, the production site and a remote standby site. Both sites host two database servers each. At any time, only one site serves client connectivity and the other acts as a hot standby. Both of the sites are set up with identical hardware and software components, including their versions. Using identical hardware and software at both sites is only a recommendation; it is not an absolute requirement to match every component in the entire solution stack between the production and remote sites. Refer to vendor-specific product compatibility guides to use specific features. Optionally, a test/dev server is available at the primary site to host a copy of production database using NX4 writable checkpoint technology.

The solution uses Celerra Replicator (V2) for asynchronously replicating the production file system to a remote Dell NX4. Celerra Replicator (V2) is available in Celerra 5.6 or newer releases.

To isolate the database network traffic from public network traffic, a dedicated virtual LAN is used. However, a physically separated LAN can also achieve this purpose.

General characteristics

The Oracle RAC 10g configuration runs Oracle Database 10g Enterprise Edition over Red Hat Enterprise Linux (RHEL 4 Update 5).

The Oracle RAC 10g for Linux on NFS configuration does the following:

- ◆ Stores Oracle database files along with a single copy of online redo log files on a dedicated NX4 file system.
- ◆ Stores archived redo log files and backed up database files on dedicated NX4 file systems.
- ◆ Stores mirrored copies of database control files in the NX4 file system with data files.
- ◆ Mirrors the Oracle Cluster Registry (OCR) and Voting Disk files.
- ◆ Uses RAID-protected NX4 file systems to satisfy the I/O demands of individual database objects.
- ◆ Stores all database files on the Dell NX4 series storage system, making server replacement relatively simple.
- ◆ Runs Oracle RAC 10g R2 Enterprise Edition x86-64 over RHEL4 Linux servers with 12 GB of physical RAM.

ORACLE 10G AND LINUX NFS ON DELL NX4

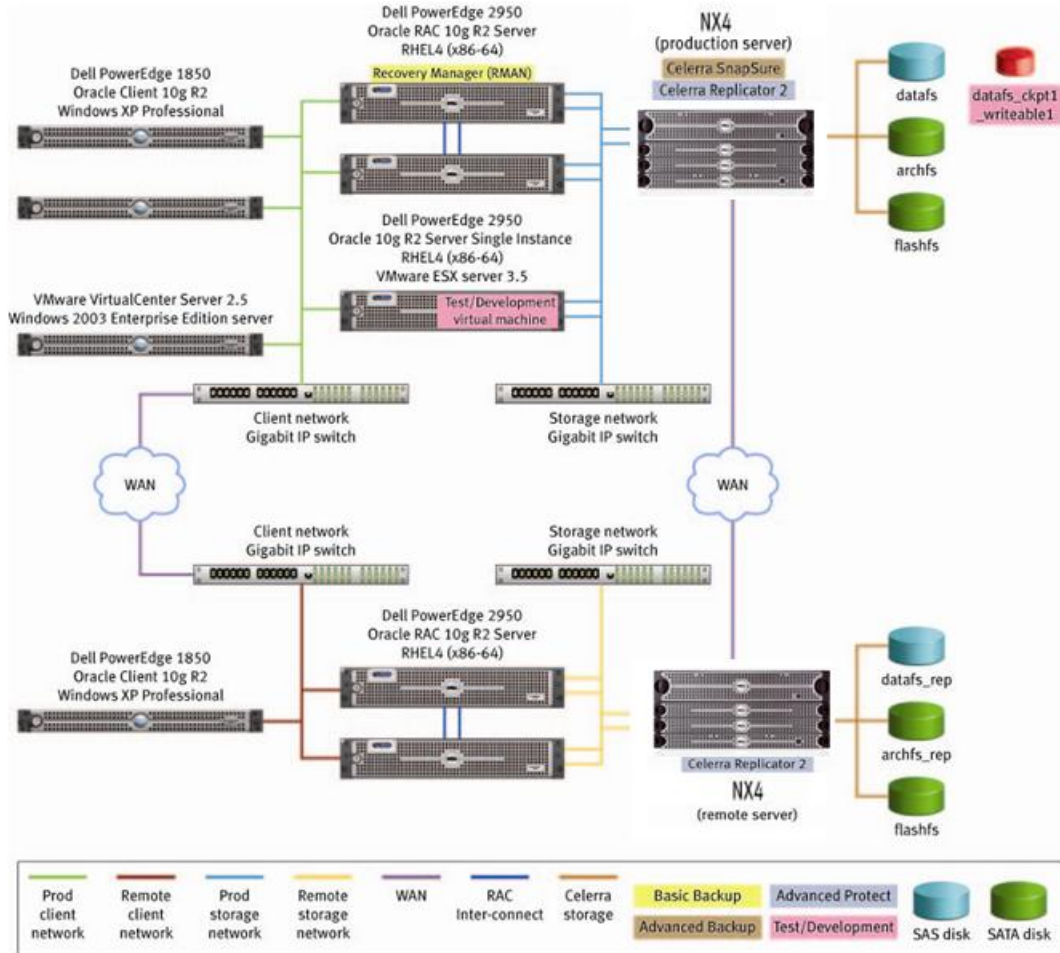


Figure 1 Oracle RAC 10g for Linux on NFS - system architecture

Network architecture

This section describes the network architecture of the validated solution.

Virtual local area networks

The validated solution uses three VLANs to segregate network traffic. This improves throughput, manageability, application separation, high availability, and security as shown in

Figure 1. The Oracle RAC 10g for Linux on NFS system architecture contains the following VLANs:

- ◆ The client/driver VLAN supports connectivity between the Oracle RAC 10g servers and the client workstations. The client VLAN also supports connectivity between the Dell NX4 and the client workstations to provide network file services to the clients. Control and management of these devices are also provided using the client network.
- ◆ The RAC interconnect VLAN supports connectivity between the Oracle RAC 10g servers for network I/O as required by Oracle Cluster Ready Services (CRS). All information sharing, such as cluster heartbeat, and database coherency data between nodes, is done over this network. Two network interface cards are configured on each Oracle RAC 10g server to the RAC Interconnect network. Link aggregation is configured on the servers to provide load balancing and port failover between the two ports for this network.
- ◆ The Storage VLAN uses the NFS protocol to provide connectivity between the Oracle servers and storage. The database servers connected to the storage VLAN have two NICs dedicated to the storage VLAN. Link aggregation is configured on the servers to provide load balancing and port failover between these two ports for this network. The two network ports used on the Storage VLAN are teamed together using NX4 advanced networking features to provide load balancing and failover functionality.

Switches

For IP switches on which the client and storage VLANs are configured, Dell recommends that these switches support Gigabit Ethernet (GbE) connections, jumbo frames, and port channeling.

Dell NX4 Data Mover ports

This section describes the network ports on the rear of a Dell NX4 Data Mover.

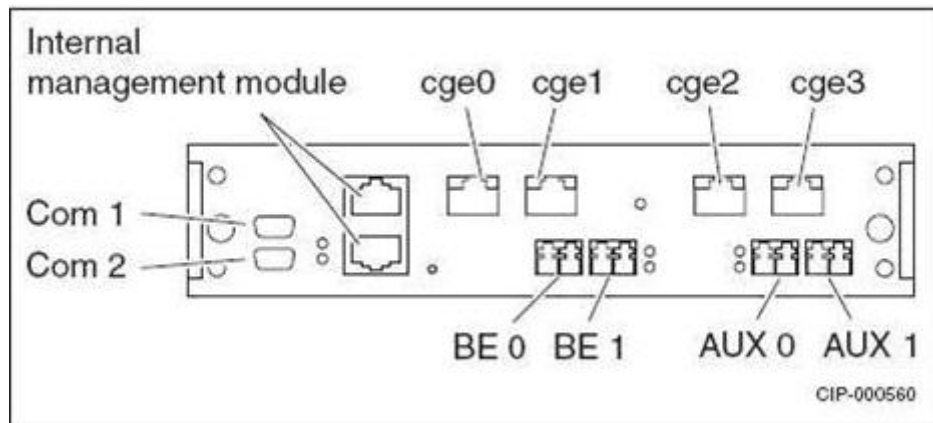


Figure 2 Dell NX4 Data Mover ports and traffic types

Ports cge0 and cge1 are aggregated using the link aggregation protocol, and are used for the database storage network. They handle all I/O that is required by the database servers to the data files, online redo log files, archived log files, control files, backup files and cluster configuration files. Ports cge2 and cge3 handle other load such as replication on the file system.

Storage architecture

The steps for setting up the storage configuration are:

1. Establish the RAID levels
2. Allocate hot spares
3. Create LUNs
4. Create disk volumes
5. Create metavolumes from disk volumes by striping or concatenating
6. Create file systems from metavolumes
7. Export the file systems to be used by NFS clients
8. Mount file systems on Oracle RAC 10g database servers
9. Enable file system replication between the production and remote NX4 Network Servers

High availability and failover

The Dell NX4 has built-in high-availability features. These HA features allow the NX4 to survive various failures without losing access to the Oracle RAC 10g database server. These features protect against the following:

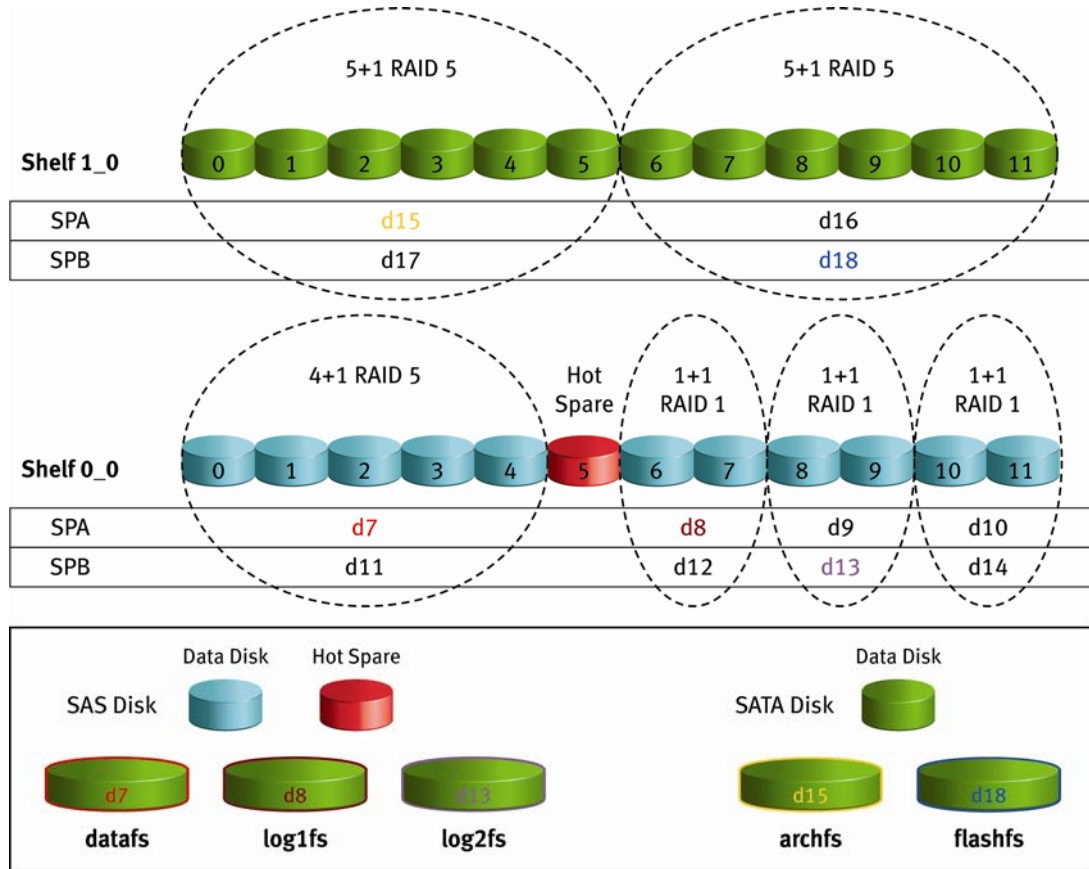
- ◆ Data Mover failure
- ◆ Data Mover port failure
- ◆ Power loss affecting a single circuit connected to the storage array
- ◆ Storage processor failure
- ◆ Disk failure

The Dell NX4 is configurable for either a single or dual Data Mover. If the single Data Mover option is chosen, the NX4 Data Mover failover feature is not available.

RAID type and RAID group configuration

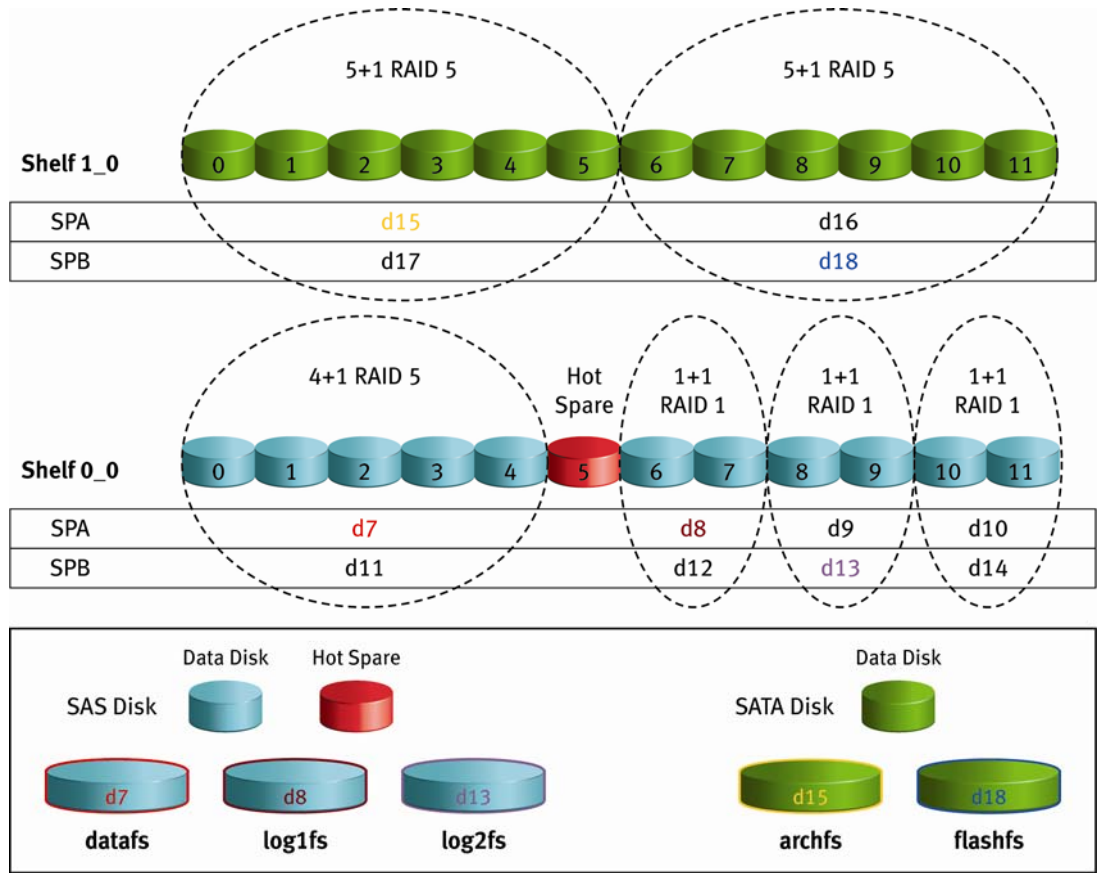
The traditional approach for NX4 storage configuration is to place Oracle database files and two copies of multiplexed redo log files on separate sets of physical disks, with the storage configured in accordance with their data access patterns. Along with the traditional approach, the current solution uses a building block (BB) approach to provision storage for database files. In the building block (BB) approach, the data files and a single copy of redo log files are placed on a single NX4 file system, for example, on the same set of physical disks. This eases the storage management and provides a consistent recoverable database point-in-time copy with advanced backup and protect solution components.

Each building block is a 5+1 RAID 5 group created on SAS drives. Depending on the configuration, a specific number of building blocks are used to create the file system that holds the database data files, redo log files, control files, OCR files and voting disk files. One shelf of SATA drives is configured with 5+1 RAID 5, as illustrated in Figure 3, Figure 4, and Figure 5. The archived log files and flashback logs are placed on SATA drives since they have lower performance requirements.



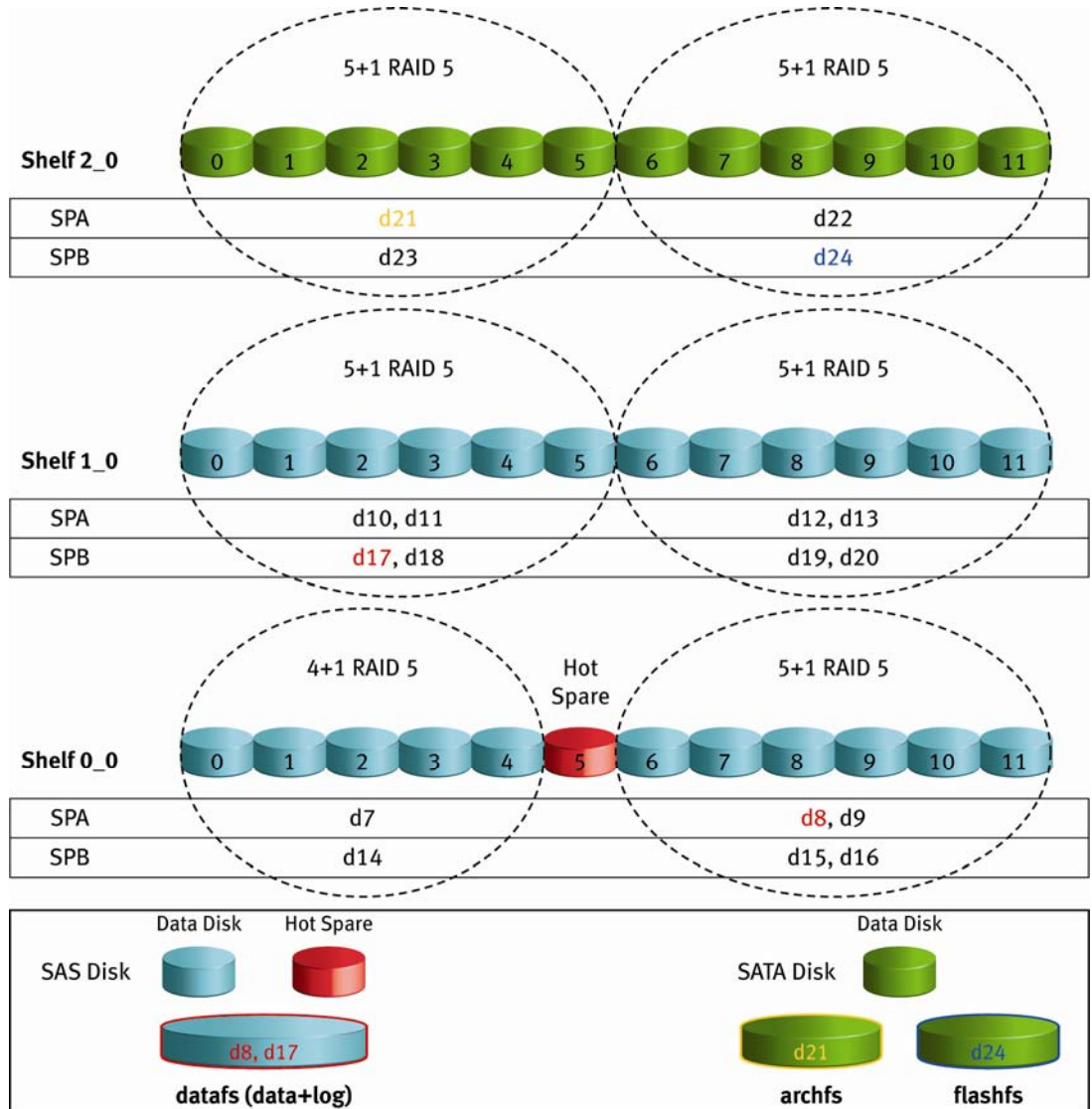
GEN-000999

Figure 3 Oracle RAC 10g – traditional configuration



GEN-000999

Figure 4 Oracle RAC 10g – one building block configuration



GEN-000997

Figure 5 Oracle RAC 10g – two building blocks configuration

The SAS drives are used to store Oracle RAC 10g data files, online redo log files, control files, OCR files and voting disk files. The SATA drives are used for archived log files and backed up database data files.

The RAID group layout for this solution was created using the following standard NX4 storage templates:

NX4_4+1R5_HS_R1/0_R1/0_R1/0

NX4_4+1R5_HS_5+1R5

NX4_5+1R5_5+1R5

Disk volume setup

After the RAID groups are created and LUNs are bound, the Dell NX4 automatically maps disk volumes to physical LUNs and the disk volumes are accessible to the Data Movers.

File systems, exports, and mount points

In the next step, the database file systems are created on a Dell NX4 network server. In our solution testing, the consolidation tests alone were performed using a traditional approach as well as with both one building block and two building blocks. All the remaining solution components were tested using a two building blocks configuration. The database file systems and the corresponding storage disk volumes used to create them are shown in [Table 3](#).

Table 3 File system layout

File system	Storage volumes
/datafs	<ul style="list-style-type: none"> • Traditional approach – uses one disk volume from the 4+1R5 RAID group of SAS shelf 0_0 • One building block approach – uses two disk volumes concatenated from the 5+1 R5 RAID group of SAS shelf 0_0 • Two building blocks approach – uses one disk volume striped from each of the two 5+1 R5 RAID groups of SAS shelves 0_0 and 1_0
/archfs	Uses the disk volume from the first 5+1 R5 group (SATA shelf 2_0)
/flashfs	Uses the disk volume from the second 5+1 R5 group (SATA shelf 2_0)

The NX4 file systems are mounted on the Data Mover and then exported with appropriate permissions.

Database server architecture

The two-node Oracle RAC 10g server is installed on Red Hat Enterprise Linux 4 (x86-64), which is running on a Dell 2950 server. This section describes the database server architecture.

Oracle RAC 10g server network architecture

Each Oracle RAC 10g server has five network interfaces. Two interfaces connect the database server to the Data Mover storage network. Two other interfaces connect the server to RAC interconnect, enabling the heartbeat and other network I/O as required by Oracle cluster-ready services. One more interface connects to the client network. Refer to [Table 4](#) for the list of the interfaces used on each database server for current solution.

Table 4 Database server network interface configuration

Interface port ID	Description
eth1	Client network
eth2	Storage network
eth3	Storage network
eth4	RAC interconnect
eth5	RAC interconnect

High availability and failover

TCP/IP provides the ability to establish redundant paths for sending I/O from a networked computer to another networked computer. The current solution uses link aggregation on the database servers between the two ports each used for the storage and the RAC interconnect network. This feature provides redundant paths that facilitate high availability and load balancing for the networked connection.

Database software

The Oracle RAC 10g software binary files were installed on the database servers' local disks. Data files, online redo log files, archived log files, flashback recovery files, temp files, OCR files and voting disk files resided on Dell NX4 file systems. The file systems were designed (in terms of the RAID level and number of disks used) to be appropriate for each type of database file. [Table 5](#) lists each file type and its location.

Table 5 Database file types and locations

File type	Location
Oracle RAC software binary files	Database server local disk
Oracle database software binary files	Database server local disk
Data files, temp files	/datafs
Online redo log files	/log1fs and /log2fs (traditional approach)
	/datafs (building block approach)
Archived log files	/archfs
Flash recovery area	/flashfs
Control files	Mirrored in /datafs
SP file	/datafs
OCR files	Mirrored in /datafs (building block approach)
	Mirrored in /datafs, /log1fs, and /log2fs (traditional approach)
Voting disks	Mirrored in /datafs (building block approach)
	Mirrored in /datafs, /log1fs, and /log2fs (traditional approach)

High availability and failover

This validated solution provides protection at the storage, network, host, and site levels.

Storage layer

The Dell NX4 network server has the option of having one or two Data Movers, which facilitates high availability and load balancing. In this solution, the two NX4 Data Movers, configured as primary and standby, together provide seamless failover capabilities for the NX4 file system storage. The RAID disk configuration on the NX4 backend, along with the disk hot spare feature, provides protection against hard disk failure.

Network layer

The advanced networking features of the Dell NX4 network server, such as fail-safe network and link aggregation, provide protection against network connection failures. The solution configuration also includes multiple NICs on the host side, a separate network infrastructure (cables, switches, routers, and so on), and separate target ports. This ensures there is no single point of network failure.

Host layer

This solution is configured with Oracle two-node RAC 10g. Due to inherent properties of the cluster, Oracle RAC provides high availability and failover capabilities between the RAC nodes. The details of which are available in the references and is outside the scope of this document.

Site level protection

The Celerra Replicator V2 feature provides a method of replicating production data to a remote disaster recovery site. Through proper design of an end-to-end environment, the required RPO and RTO can be achieved in case the production site goes down.

Hardware and software resources

Hardware resources

[Table 6](#) lists the hardware resources for the Oracle RAC 10g for Linux on NFS solution.

Table 6 Hardware specifications

Equipment	Quantity	Configuration
Dell 2950 (EM64) server (Database server)	Four (Two for production site, two for remote site)	<ul style="list-style-type: none"> Two 3.0 GHz Intel Xeon dual-core processors 12 GB of memory One 73 GB 10k internal SCSI drive Two onboard 10/100/1000 Mb Ethernet NICs Two additional dual port 10/100/1000 Mb Ethernet NICs
Dell 2950 server (VMware ESX Server)	One	<ul style="list-style-type: none"> Two 3.0 GHz Intel Xeon dual-core processors 16 GB of memory One 73 GB 10k internal SCSI drive Two onboard 10/100/1000 Mb Ethernet NICs
Dell 1850	Three (Two for database clients, One for VMware virtual center)	<ul style="list-style-type: none"> Two 3.0 GHz Intel Xeon single-core processors 4 GB of memory One 73 GB 10k internal SCSI drive Two onboard 10/100/1000 Mb Ethernet NICs
Gigabit Ethernet switch	One for each VLAN (one for client network, one for RAC interconnect, one for storage network)	<ul style="list-style-type: none"> VLAN support Optional jumbo frame support LACP and/or EtherChannel support
Dell NX4 network server	Two (one for production site, one for remote site)	<ul style="list-style-type: none"> Two Data Movers DART 5.6 Four GbE network connections per Data Mover Two SAS shelves (Twelve 146 GB 15k rpm SAS disks on each shelf) One SATA shelf (Twelve 750 GB 7.2k rpm SATA disks) One Control Station

Software resources

[Table 7](#) lists the software resources for the Oracle RAC 10g for Linux on NFS solution.

Table 7 Software specifications

Software title	Number of licenses
Celerra Manager Advanced Edition	One per NX4 server
Celerra Replicator V2	One per NX4 server
Red Hat Enterprise Linux (x86-64)	One per database server
Oracle RAC 10g R2 Enterprise Edition for Linux(x86-64)	One per database server
VMware ESX Server 3.5	One for ESX Server
VMware VirtualCenter Management Server 2.5	One for VMware VirtualCenter
Quest Benchmark Factory 5.5	One per database client

Chapter 4 Solution Best Practices

This chapter presents these topics:

Solution architecture best practices	30
Storage setup and configuration	33
Network setup and configuration	36
Database Server setup and configuration	37
Linux setup and configuration.....	38
Oracle database setup and configuration	39
Backup, recovery and protect setup and configuration	41
Virtual test/dev environment configuration.....	42

Solution architecture best practices

This section discusses the best practices for running Oracle RAC 10g Enterprise Edition x86-64 on a RHEL4U5 Linux server with Dell NX4 network server. The topics covered include setup and configuration of:

- ◆ Storage
- ◆ Network
- ◆ Database server hardware and BIOS
- ◆ Linux operating system
- ◆ Oracle software install
- ◆ Database parameters and settings
- ◆ Backup, recovery, and protection
- ◆ Virtual test/dev environment

Oracle RAC 10g database performance tuning is beyond the scope of this guide. The *Oracle Database Performance Tuning Guide 10g Release 2 (10.2)* provides more information on this topic. This guide is available from the Oracle website.

Solution validation and performance testing overview

The Oracle RAC 10g Linux on Dell NX4 NFS solution has been validated by Dell NAS Solutions Engineering. The reference architecture was tested for functionality and iterative performance tuning. TPC-C benchmark performance tests were run using Quest Benchmark Factory 5.5. The functional testing phase of the solution validated the following:

- ◆ Proper database operation
- ◆ Database integrity
- ◆ Database recoverability
- ◆ Database protection

The performance tuning phase of the solution tested Oracle database transaction response times under an increasing TPC-C user load. The gating metric for the test was an average response time of 2 seconds. The test was considered as failed once the TPC-C transaction response time exceeded the 2-second limit.

Iterations of this test procedure were performed to isolate and tune various system and Oracle settings. Refer to Table 11 on page 40 for more information about these settings.

With a two-node RAC configuration using a two-building block approach, the performance optimization tests yielded a maximum user load of approximately 3,600 TPC-C users and 187 transactions per second in a tuned environment.

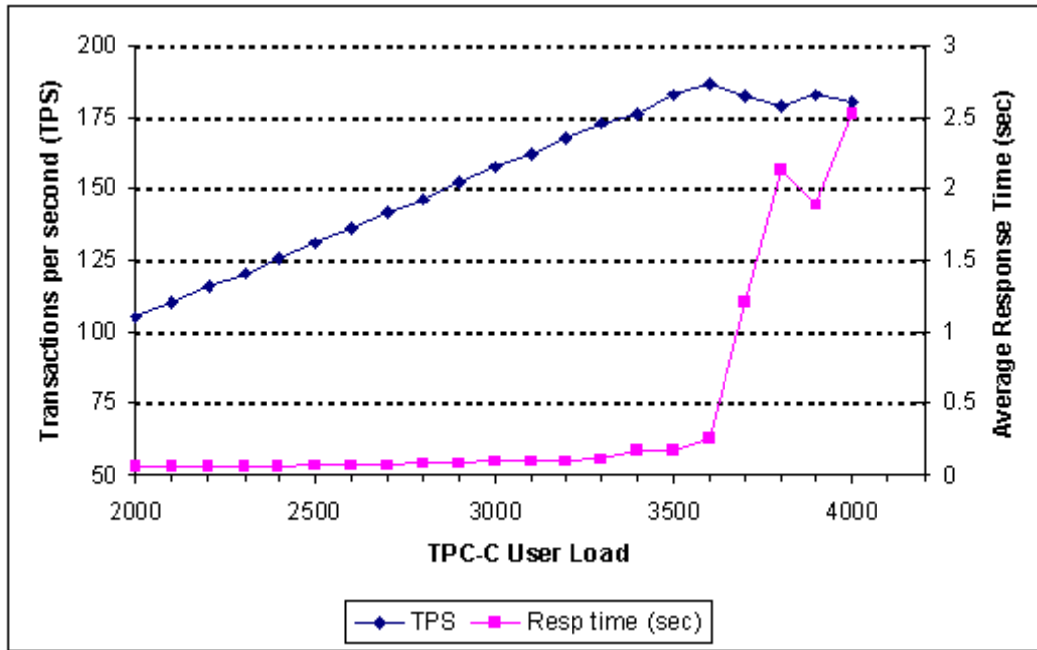


Figure 6 TPC-C user load scaling

The Oracle RAC 10g database servers experienced a modest load on the CPU, while the Data Mover CPU on the NX4 showed only minor inflection. The Data Mover CPU shows plenty of headroom for scalability, and the ability to service other processes such as file serving. CPU utilization is demonstrated in Figure 7:

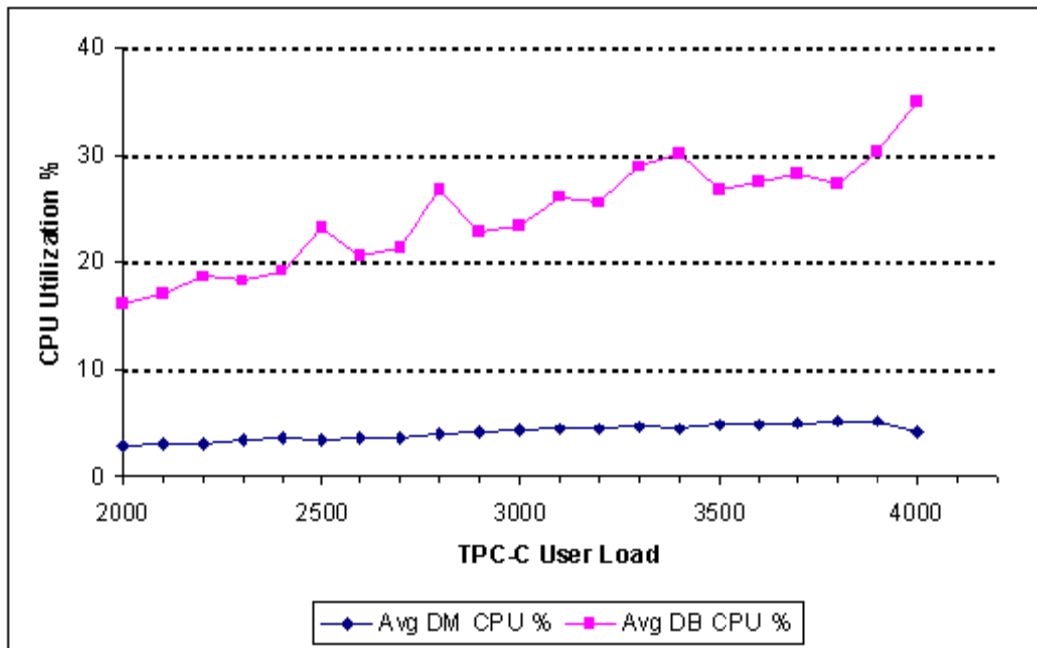


Figure 7 CPU utilization chart

From a one-node configuration to a two-node configuration with the two building block approach, Oracle RAC 10g database showed smooth scaling as shown in [Figure 8](#).

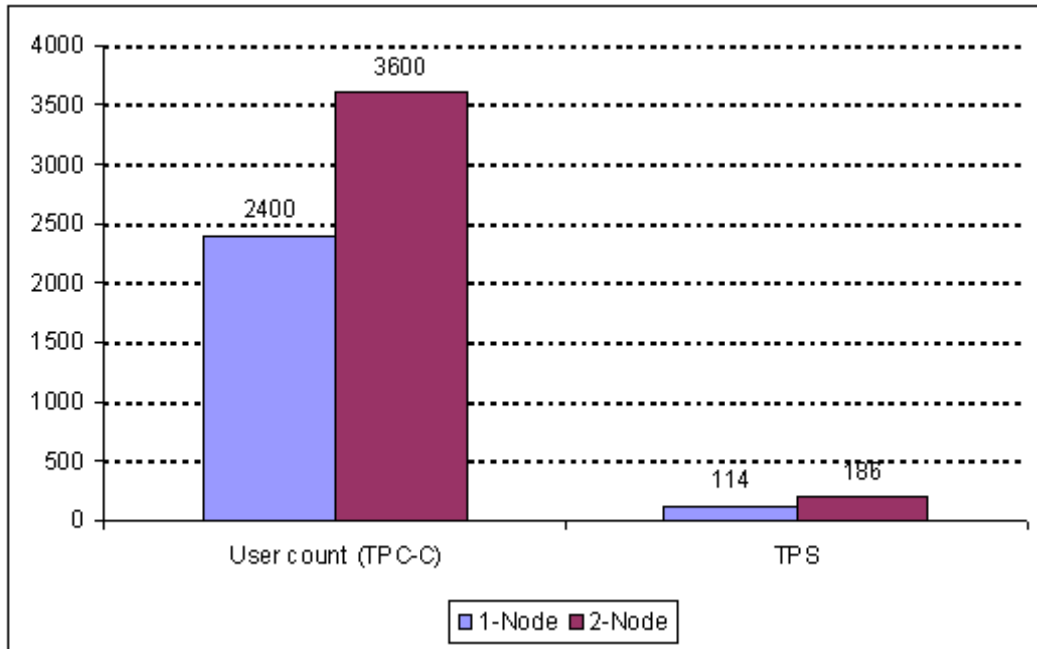


Figure 8 RAC scaling

This solution was tested using a traditional approach along with one and two building block approach. [Figure 9](#) summarizes the test results in terms of transactions per second (TPS).

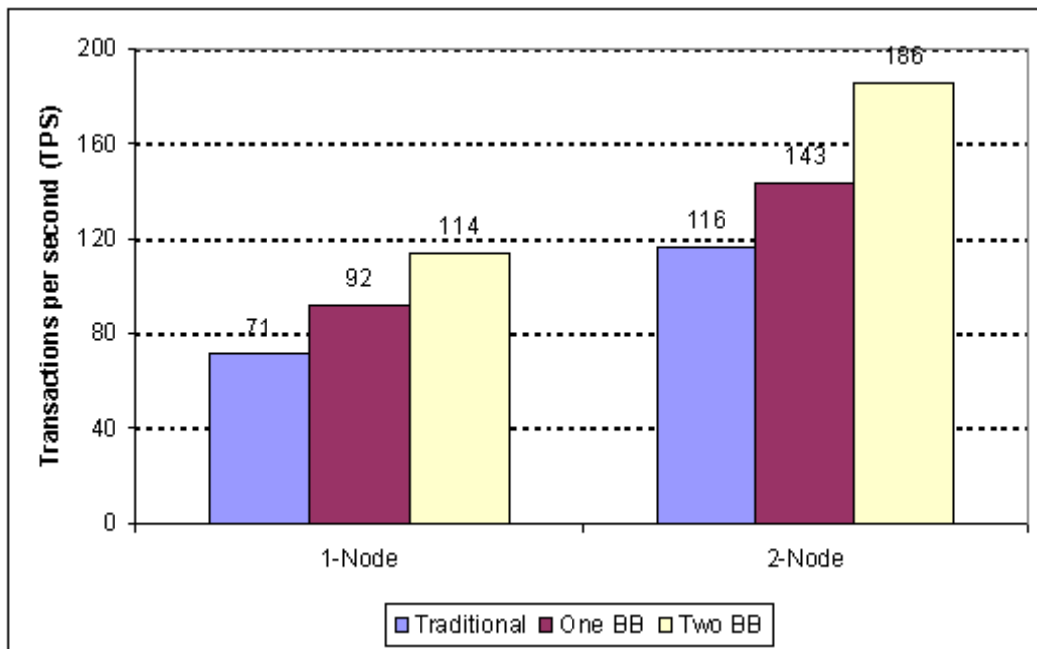


Figure 9 Tested configurations – TPS

Storage setup and configuration

General disk drive recommendations

This section contains general recommendations for disk drive settings:

- ◆ Drives with higher revolutions per minute (rpm) provide higher overall random-access throughput and shorter response times than drives with slower rpms. For better performance, higher-rpm drives are recommended.
- ◆ Because of significantly better performance, Serial Attached SCSI (SAS) drives are always recommended for storing data files and online redo log files, instead of SATA drives.
- ◆ Serial Advanced Technology Attached (SATA) drives have slower response rotational speed and moderate performance with random I/O. However, they are less expensive than SAS drives for the same or similar capacity. SATA drives are therefore the best option for storing archived redo logs and backup files.

Database file storage

Refer to *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide 10g Release 2 (10.2) for Linux* for information on placing the shared database files such as data files, online redo log files, archived log files, and flashback recovery files in the cluster on NFS mounts for better manageability. This document is available from the Oracle website,

<http://www.oracle.com>. With the NX4 network server, Dell recommends the practice as shown in [Table 8](#) to place the Oracle database files.

Table 8 Database file storage recommendations

NX4 file system	Contents	RAID level	Disk type
datafs	Data files, redo log files, OCR files and voting disk	5	SAS
archfs	Archived log files	5	SATA
flashfs	Flashback recovery area	5	SATA

Best practices for file system design dictate that a file system should consist entirely of volumes that are all of the same RAID type and that consist of the same number and type of component spindles. Therefore, Dell does not recommend mixing any of the following within a single database file system:

- ◆ RAID levels
- ◆ Disk types
- ◆ Disk rotational speeds

Volume management

With the current Oracle 10g RAC on Dell NX4 solution, manual volume management has been used to create the database file systems and to facilitate the usage of a building block approach for consolidation testing. The NX4 disk volumes selected for striping are with LUNs that alternate ownership between the two storage processors of the back-end storage array.

NX4 stripe size

Dell recommends a stripe size of 32 KB for all types of database workloads. The default stripe size for all the file systems on SAS disks should be 32 KB. Similarly, the recommended stripe size for the file systems on SATA disks (archive and flash) should be 256 KB.

Data Mover parameter setup

Noprefetch

Dell recommends that you turn off the file system read prefetching for an online transaction processing (OLTP) workload. Leave it on for a DSS workload. Prefetch will waste I/Os in an OLTP environment since few, if any, sequential I/Os are performed. In a DSS workload, you should turn on the read prefetch.

To turn off the read prefetch mechanism for a file system, type:

```
$server_mount <movername> -option <options> noprefetch <fs_name>  
<mount_point>
```

Uncached

This setting allows well-formed writes (for example, multiples of a disk block and disk block aligned) to be sent directly to the disk without being cached on the server. Dell testing shows significant improvement in the performance by turning off file system write caching for OLTP workload.

To turn off the write cache mechanism for a file system, type:

```
$server_mount <movername> -option <options> uncached <fs_name>  
<mount_point>
```

NFS threads

Dell recommends that you use the default NFS thread count of 256 for optimal performance. Please do not set this to a value lower than 32 or higher than 512.

File.asyncthreshold

Dell recommends that you use the default value of 32 for the parameter *file.asyncthreshold*.

NFS mount point parameters

For optimal reliability and performance, the NFS client options in [Table 9](#) are recommended. The mount options are listed in the `/etc/fstab` file on Oracle RAC 10g database servers.

Table 9 NFS mount point options on database servers

Option	Syntax	Recommended	Description
Hard mount	hard	Always	With this option, the NFS file handles are kept intact when the NFS server does not respond. When the NFS server responds, all the open file handles resume, and do not need to be closed and reopened by restarting the application. This option is required for Data Mover failover to occur transparently without having to restart the Oracle instance.
NFS protocol version	Vers=3	Always	This option sets the NFS version to be used. Version 3 is recommended.
TCP	Proto=tcp	Always	With this option, all the NFS and RPC requests will be transferred over a connection-oriented protocol. This is required for reliable network transport.
Background	Bg	Always	This setting enables client attempts to connect in the background if the connection fails.
No interrupt	nointr	Always	This toggle allows or disallows client keyboard interruptions to kill a hung or failed process on a failed hard-mounted file system.
Read size and write size	rsize=32768, wsize=32768	Always	This option sets the number of bytes NFS uses when reading or writing files from an NFS server. The default value is dependent on the kernel. However, throughput can be improved greatly by setting rsize/wsize= 32768
No auto	noauto	Only for backup/utility file systems	This setting disables automatic mounting of the file system on bootup. This is useful for file systems that are infrequently used (for example, stage file systems).
ac timeo	actimeo=0	RAC only	This setting disables attribute caching for regular files and directories.
Timeout	timeo=600	Always	This sets the time (in tenths of a second) the NFS client waits for the request to complete.

Load distribution

For Oracle database tablespaces with heavy I/O workloads consisting of concurrent reads and writes, Dell recommends spreading the I/O load across multiple spindles. [Figure 5](#) on page 22 illustrates the two NX4 disk volumes that are used in which to place the Oracle database data files. This effectively distributes the I/O load across 12 SAS spindles.

High availability

The Data Mover failover capability is a key feature unique to the NX4 Network Server. This feature offers redundancy at the file server level, allowing continuous data access. It also helps in building a fault-resilient Oracle RAC 10g architecture.

Dell recommends that you set the Data Mover failover policy to auto. This allows the Control Station to immediately fail the Data Mover over to its standby in the event of hardware or software failure.

Upon failover, the standby Data Mover assumes the following functions from the faulted Data Mover:

- ◆ **Network identity:** The IP and MAC addresses of all its NICs
- ◆ **Storage identity:** The file systems that the faulted Data Mover controlled
- ◆ **Service identity:** The shares and exports that the faulted Data Mover controlled

This ensures continuous access to the NX4 file systems for the database. The database servers do not see any significant interruption in I/O.

Data Mover failover occurs if any of the following conditions exists:

- ◆ Failure (operation below the configured threshold) of both internal network interfaces by the lack of a heartbeat (Data Mover timeout)
- ◆ Power failure within the Data Mover (unlikely, as the Data Mover is typically wired into the same power supply as the entire array)
- ◆ Software panic due to exception or memory error
- ◆ Data Mover hang

Data Mover failover does not occur under the following conditions:

- ◆ Removing a Data Mover from its slot
- ◆ Manually rebooting a Data Mover

Since manual rebooting of Data Mover does not initiate a failover, Dell recommends that you initiate a manual failover before taking down a Data Mover for maintenance.

The synchronization services component (CSS) of Oracle Clusterware maintains two heartbeat mechanisms:

- ◆ The disk heartbeat to the voting disk
- ◆ The network heartbeat across the RAC interconnect, which establishes and confirms valid node membership in the cluster

Both of these heartbeat mechanisms have an associated timeout value. For more information on Oracle Clusterware MissCount and DiskTimeout parameters, see Oracle MetaLink Note 2994430.1.

You should leave the network heartbeat parameter misscount to the default of 60 seconds and set the disk heartbeat parameter disktimeout to 600 seconds. These settings will ensure that the RAC nodes do not evict when the active Data Mover fails over to its standby, or when the active Data Mover reboots.

Network setup and configuration

This section contains Dell recommendations to set up and configure the network for optimal performance.

Gigabit connection

Use Gigabit Ethernet for the network connections between database servers and the NX4 Network Server. Avoid the use of 100BaseT. Using Gigabit Ethernet for the RAC interconnect is preferable.

Virtual local area networks

Use virtual local area networks (VLANs) to provide better throughput, manageability, application separation, high availability, and security. [Table 10](#) describes the three VLANs used for this solution.

Table 10 Oracle RAC 10g solution VLANs

VLAN ID	Description	CRS setting
1	Client network	Public
2	RAC interconnect	Private
3	Storage	Do not use

Refer to the Network architecture section for more information.

Network port configuration

Spread the database I/O over two of the four available NX4 network ports. Refer to the [Network architecture](#) section for recommended configuration. In addition, set the speed and duplex settings to auto on all ports. This is one of the single most common (and easily resolvable) performance issues observed today.

Network security

Place the NX4 Network Server Data Movers and the Oracle RAC 10g database servers in private segregated storage networks to isolate the traffic between them. The network in which the external interface of the Control Station resides should also be on a segregated subnet, secured by the firewall, since the Control Station is the most important administrative interface to the NX4 Network Server.

Any other security policies implemented in your organization should be employed to secure the environment.

Jumbo frames

Maximum Transfer Unit (MTU) sizes of greater than 1,500 bytes are referred to as *jumbo frames*. Jumbo frames require Gigabit Ethernet across the entire network infrastructure (server, switches, and database servers).

Whenever possible, Dell recommends the use of jumbo frames on all legs of the storage network. For Oracle RAC 10g installations, jumbo frames are recommended for the private RAC interconnects to boost throughput as well as to possibly lower CPU utilization due to the software overhead of the bonding devices. Jumbo frames increase the device MTU size to a larger value (typically 9,000 bytes). NX4 Data Movers support MTU sizes of up to 9,000 bytes.

Typical Oracle database environments transfer data in 8 KB and 32 KB block sizes and require multiple 1,500 byte frames per database I/O, while using an MTU size of 1,500 bytes. Using jumbo frames reduces the number of frames needed for every large I/O request and, in turn, reduces the host CPU needed to generate a large number of interrupts for each application I/O. The benefit of jumbo frames is primarily a complex function of the workload I/O sizes, network utilization, and NX4 Data Mover CPU utilization, so it is not easy to predict.

Database server setup and configuration

Server BIOS

Regardless of your server vendor and architecture, you should monitor the BIOS version shipped with your system and determine if it is the latest production version supported by your vendor. Frequently, it is not the latest version. If that is the case, we recommend flashing the BIOS.

Hyperthreading

Intel hyperthreading technology allows multithreaded operating systems to view a single physical processor as if it were two logical processors. A processor that incorporates this technology shares CPU resources among multiple threads. In theory, this enables faster enterprise server response times and provides additional CPU processing power to handle larger workloads. As a result, server performance is likely to improve. Dell testing, however, shows that the performance with hyperthreading was worse than without hyperthreading. For this

reason, Dell recommends disabling this feature. Use the BIOS configuration menu to disable hyperthreading. Please refer to your server vendor documentation for instructions.

Memory

Since Oracle workload is always memory intensive, Dell recommends that you configure the system with the maximum amount of memory feasible to meet your scalability and performance needs. Refer to the database server documentation to determine the total number of memory slots your database server has, and the number and density of memory modules that you can install.

Our testing on the Oracle RAC 10g Linux on NX4 NFS solution has been performed using 12 GB memory on each of the database servers.

Shared memory

Oracle uses shared memory segments for the Shared Global Area (SGA), which is an area of memory that is shared by Oracle processes. The size of the SGA has a significant impact on database performance.

Our Oracle RAC 10g testing was done with database servers using 8 GB of SGA.

SHMMAX setting

This parameter defines the maximum size in bytes of a single shared memory segment that a Linux process can allocate in its virtual address space. Since the SGA is comprised of shared memory, SHMMAX can potentially limit the size of the SGA. SHMMAX should be slightly larger than the SGA size.

As the SGA size was set to 8 GB, SHMMAX was set to 10 GB as shown:

```
kernel.shmmax = 10737418240
```

SHMMNI setting

This parameter sets the systemwide maximum number of shared memory segments. Oracle recommends SHMMNI to be at least 4096 for Oracle 10g as shown:

```
kernel.shmmni = 4096
```

Huge pages setting

This parameter specifies the number of physically contiguous, large memory pages that will be allocated and pinned in RAM for shared memory segments like the Oracle SGA. Huge pages must be large enough to accommodate the entire SGA. Unused huge pages will not be available for use other than for shared memory allocations, even if the system runs out of memory and starts swapping. For this reason huge pages must be tuned carefully and set correctly.

Huge pages can create a very significant performance improvement for the Oracle RAC 10g database servers. Set the huge pages parameters so that the entire SGA can be allocated with huge pages. These settings are in `/etc/sysctl.conf`. Refer to Oracle Metalink 361323.1 for more information on enabling and tuning huge pages.

Our testing on the Oracle RAC 10g Linux on NX4 NFS solution has been performed using 4,097 huge pages each of size 2 MB on each of the database servers.

Linux setup and configuration

Kickstart provides a way for users to automate a Red Hat Enterprise Linux installation. This is particularly critical in RAC environments where the OS configuration should be identical, and the required packages are more specific. Using kickstart, a single file can be created containing the answers to all the questions that would normally be asked during a Linux installation. These files can be kept on a single server system and read by individual database servers during the installation, thus creating a consistent, repeatable Linux install.

Refer to the Oracle documentation on installing Oracle Database 10g Release 2 on Linux x86-64 for detailed instructions and recommendations.

The steps for kickstart installation are as follows:

1. Create a kickstart file.
2. Create a boot media with the kickstart file or make the kickstart file available on the network.
3. Make the installation tree available.
4. Start the kickstart installation.

[Sample ks.cfg](#) provides a sample ks.cfg file that you can use. This file was used in Dell testing. For a clean, trouble-free Oracle Clusterware install, use these packages exclusively for an Oracle RAC 10g installation.

Note: On the Intel EM64T platform, both the 32-bit and 64-bit versions of libaio are required. In order to install this rpm successfully on this platform, the following procedure is required (assuming the current working directory contains both the 32-bit and 64-bit versions of this rpm):

```
rpm -e --nodeps libaio
rpm -Uvh libaio*rpm
```

Before beginning Oracle software installation, the Linux operating system should be configured to maximize performance for the database servers.

Virtual memory

Oracle uses virtual memory. Although the paging file(s) should not be consistently used, it is far worse to run short of virtual memory when you need it on a temporary basis.

Set the virtual memory to one to four times of physical RAM installed in the server. Try to place the page file on a different physical internal disk than the disk where operating system is installed. If at all possible, split the paging file into multiple files on multiple physical devices. This encourages parallel access to virtual memory, and increases performance.

Oracle memory structure

Set the Oracle System Global Area (SGA) size to an optimal value suitable for your environment. One of the worst performance pitfalls is to set the Oracle memory structures too large. In many cases, the SGA or Program Global Area (PGA) may be partially swapped to disk (using virtual memory). Since disk I/O is several orders of magnitude slower than main memory, this has a drastic effect on performance.

Static IP address

Configure all network interfaces to use a static IP address instead of a DHCP server assigned IP address.

Oracle database setup and configuration

Oracle database file placement

We discourage placing Oracle database files for different instances (multiple databases running on the same production servers) on the same set of file systems. In such cases, you should create a separate set of NX4 file systems for each database. It is also recommended to place the cluster configuration files such as OCR files and voting disk files on a file system separate from those on which database files are placed. This provides the flexibility in using NX4 advanced features such as SnapSure, Replicator V2, and others.

Oracle database initialization parameters

To configure the Oracle instance for optimal performance with the NX4 Network Server, we recommend the initialization options in [Table 11](#) contained in the *spfile* or *init{ORACLE_SID}.ora* file for the Oracle instance.

Table 11 Database parameter initialization options

Parameter	Syntax and description
Database block size	<p>DB_BLOCK_SIZE=n</p> <p>For best database performance, DB_BLOCK_SIZE should be a multiple of the OS block size. For example, if the operating system page size is 4096, DB_BLOCK_SIZE=4096*n</p>
File system I/O	<p>FILESYSTEMIO_OPTIONS=setall</p> <p>The options available for this parameter are:</p> <p>directio: This setting enables direct I/O. Direct I/O is a feature available in modern file systems that delivers data directly to the application without caching in the file system buffer cache. Direct I/O preserves file system semantics and reduces the CPU overhead by decreasing the kernel code path execution. I/O requests are directly passed to the network stack, bypassing some code layers. Direct I/O is a very beneficial feature to Oracle's log writer, both in terms of throughput and latency.</p> <p>asynch: This setting optimizes the concurrency of queuing multiple I/O requests to the storage device, allowing the application code to continue processing until the point where it simply must wait for the I/O requests to complete.</p> <p>setall: This option turns on both direct I/O and asynch I/O. This is the recommended setting.</p>
Disk async I/O	<p>DISK_ASYNC_IO=true</p> <p>This parameter controls whether I/O to data files, control files, and redo log files is asynchronous or not. Async I/O is now recommended on all the storage protocols.</p>
Multiple database writer processes	<p>DB_WRITER_PROCESSES=1</p> <p>Dell's testing with the configuration specified in this document showed that database performance degraded marginally if multiple database writer processes existed.</p>
Shared Servers	<p>SHARED_SERVERS =m and</p> <p>DISPATCHERS=(PROTOCOL=TCP) (DISPATCHERS=n)</p> <p>Use this mode when a large number of users need to connect to the database. It is also useful when database memory is limited or when better performance is needed. The value for "m" and "n" will vary depending on the environment.</p>
Multi-block read count	<p>DB_FILE_MULTIBLOCK_READ_COUNT= n</p> <p>This parameter determines the maximum number of database blocks read in one I/O during a full table scan. The number of database bytes read is calculated by multiplying the DB_BLOCK_SIZE and DB_FILE_MULTIBLOCK_READ_COUNT. The setting of this parameter can reduce the number of I/O calls required for a full table scan, thus improving performance.</p> <p>Increasing this value may improve performance for databases that perform many full table scans, but degrade performance for OLTP databases where full table scans are seldom (if ever) performed. Setting this value to a multiple of file system block size limits the amount of fragmentation that occurs in the I/O subsystem. This parameter is specified in DB Blocks and file system block size settings are in KB, so adjust as required. Dell recommends that DB_FILE_MULTIBLOCK_READ_COUNT be set between 1 and 4 for an OLTP database and between 16 and 32 for DSS.</p>
Open cursors	<p>OPEN_CURSORS=1000</p> <p>This parameter limits the maximum number of cursors (active SQL statements) for each session. The setting is application-dependent.</p>

Recommendation for Oracle database files

Server parameter file

Dell recommends placing this file on a disk drive that stores data files. The server parameter file (spfile) is required to recover the database at the remote site in case of a production database failure.

Control files

Dell recommends that when you create the control file, allow for growth by setting MAXINSTANCES, MAXDATAFILES, MAXLOGFILES, and MAXLOGMEMBERS to high values.

Dell recommends that your database has a minimum of two control files located on separate physical disks. One way to multiplex your control files is to store a control file copy on every disk drive that stores members of the redo log groups, if the redo log files are multiplexed. Hence, with a traditional approach, the multiplexed control files are placed on separate disk drives; whereas with building block approach, they are placed on the same physical disks.

Storing a copy of a control file on data files location is absolutely required for this solution. This ensures that with advanced protect solution component configured, a copy of the control file is available at the remote site for recovery in case the production site is not available.

Online and archived redo log files

Dell recommends that you run a mission-critical, production database in ARCHIVELOG mode. Dell also recommends that you multiplex your redo log files for these databases. Loss of online redo log files could result in failure of the database being able to recover. The best practice to multiplex your online redo log files is to place members of a redo log group on different disks.

To simplify the design and placement of database files on NX4 file systems, the Oracle RAC 10g on NX4 solution follows a building block approach, where the data files and a single copy of redo log files are placed on the same NX4 file system. For details, refer to [Figure 5](#) on page 22.

Backup, recovery, and protect setup and configuration

Dell recommends using the storage-based backup, restore and protect features that are light-weighted operations and not consuming any database server CPU or I/O channel. The best practice for the backup of Oracle RAC 10g is to perform approximately six logical backups per day on four-hour intervals, using Celerra SnapSure.

Taking logical storage backups alone is not enough to protect the database from all risks. Physical storage backups are also required to protect the database against double disk failures and other hardware failures at the storage layer. So the recommendation is to create one physical backup per day by simply copying the files from the logical backup to a different low-cost media such as SATA drives.

These logical backups can be further catalogued to an RMAN repository. Doing so allows using the RMAN restore and recovery commands at the lower granular level, such as block recovery.

For the purpose of recovering the Oracle database in case of a disaster at the production site, you must replicate the NX4 file systems containing data files, archived log files, control files, and the parameter file. The point in time to which the database can be recovered depends on various factors including:

- ◆ Online redo and archive log write intervals
- ◆ Effect of caching at various levels
- ◆ Replication session refresh frequency
- ◆ Network bandwidth between production and remote sites
- ◆ State of production system at the point of failure

By proper design and tuning of the environment (that are beyond the scope of this document), the reasonable recovery point objectives (RPO) can be achieved.

Virtual test/dev environment configuration

Many midsize enterprise customers require the feature of creating a writeable copy of the production database towards testing and development purposes. The process of provisioning this copy must create minimal, if any, impact on the production database server in terms of the performance. Absolutely no downtime can be tolerated. Further, the test/dev database created on a virtual machine will provide additional inherent advantages of virtualization such as consolidation, flexible migration, cloning, and others.

The virtual test/dev solution documented in the next chapter provides this feature using Celerra SnapSure writeable checkpoints. The checkpoints are taken by placing the database in hot backup mode, thereby providing no downtime on the production database. As the NX4 file system checkpoint operation is almost instantaneous, the database is placed in hot backup mode for a brief duration and thereby the production database has a minimal impact with the process.

A reasonable load can be placed on the test/dev database, considering the overload of checkpoints on the base NX4 file systems.

Chapter 5 Solution Applied Technologies

This chapter presents these topics:

Solution applied technologies	43
Physical backup and recovery using Oracle Recovery Manager (RMAN)	45
Logical backup and recovery using Celerra SnapSure	47
Advanced protect and recovery using Celerra Replicator (V2).....	48
Virtual test/dev solution using Celerra SnapSure.....	52

Solution applied technologies

In this chapter, many steps are listed in the form of what you type on the command line. This is referred to as a coding listing.

For example:

```
SQL>STARTUP DATABASE NOMOUNT;
```

Note the following typographic conventions for code listings:

- Linux OS commands and NX4 Control Station commands are shown in **bold** and lowercase.
- The SQL and RMAN commands are shown in **bold** and uppercase.
- Unless explicitly mentioned, all the Linux commands run on first node of the two-node RAC configuration.
- The *variable* indicating specific settings that need to be replaced according to your working environment are shown in **bold** and *italic*.

Refer to the additional details provided in [Table 12](#) for illustrating backup, protect, restore, and recovery procedures described in this chapter.

Table 12 Parameter and descriptions

Variable	Description
source_celerra (10.6.120.177)	Hostname and IP address of production NX4 Network Server
remote_celerra (10.6.120.197)	Hostname and IP address of remote NX4 Network Server
Server_2 (10.6.118.159)	Data Mover name and replication IP address of production NX4 Network Server
Server_3 (10.6.118.186)	Data Mover name and replication IP address of remote NX4 Network Server

Variable	Description
clarsas_archive clarata_archive	SavVol storage pools used by NX4 file system replication sessions
source_remote	DataMover interconnect name from production to remote NX4
remote_source	DataMover interconnect name from remote to production NX4
datafs_replica	/datafs file system replication session name
archfs_replica	/archfs file system replication session name
Production RAC database SID	MTERAC2
Database Data files path	/u02/oradata/mterac2/
Database SP file	/u02/oradata/mterac2/spfilemterac2.ora
Database control files	/u02/oradata/mterac2/control01.ctl /u02/oradata/mterac2/control02.ctl /u02/oradata/mterac2/control03.ctl
Online redo log files path	/u03/oradata/mterac2/

Physical backup and recovery using Oracle Recovery Manager (RMAN)

A complete high availability and disaster recovery strategy requires dependable data backup, restore, and recovery procedures. Oracle Recovery Manager (RMAN), a command-line and Enterprise Manager-based tool, is the Oracle-preferred method for efficiently backing up and recovering an Oracle database. Refer to *Oracle Database Backup and Recovery Basics 10g Release 2 (10.2)* for more information. This recovery reference is available from Oracle at <http://www.oracle.com>.

In the testing environment described in this solution guide, database backup and recovery were performed while running TPC-C workload against the database to study the performance impact. The observations as follows;

- ◆ **Full backup:** The full database backup performed at the achievable peak user load that did not impact the performance of the database.
- ◆ **Incremental backup:** The incremental backup strategy involves two steps. You need to perform a Level 0 backup, followed by an incremental level 1 backup at a regular interval of time. The Level 0 backup did not impact the database performance; however, the incremental backup did impact database performance for a brief period.
- ◆ **Restore and recover:** The complete recovery of the database can be performed only in the offline mode, when the database is in shutdown state. In the test environment, the database recovery from full backup and from incremental backup took almost the same time.

Dell recommends setting the number of RMAN channels to four during backup and restore operations. Dell testing has confirmed that this setting improved the database backup and restore window using RMAN as compared with the default setting (one channel).

The following commands are used towards physical backup and recovery using Oracle 10g Recovery Manager. It is assumed that the RMAN parameter settings are configured as required.

Full backup

1. Connect to RMAN.

```
$ rman target /
```

2. Back up the entire database as an image copy.

```
RMAN> BACKUP AS COPY DATABASE;
```

Full restore and recovery

1. Stop the database.
\$ srvctl stop database -d mterac2
2. Restart the database in mount mode.
\$ srvctl start database -d mterac2 -o mount
3. Connect to RMAN.
\$ rman target /
4. Restore database.
RMAN> RESTORE DATABASE;
5. Completely recover the database using archived logs and online redo log files. Make sure that the archived logs are available for database recovery.
RMAN> RECOVER DATABASE;
6. Open the database in read-write mode.
\$ sqlplus / as sysdba
SQL> ALTER DATABASE OPEN;
7. Start the second database instance.
\$ srvctl start instance -d mterac2 -i mterac22

Incremental backup

1. Connect to RMAN.
\$ rman target /
2. Back up the incremental level 0 copy of the entire database as an image copy.
RMAN> BACKUP AS COPY INCREMENTAL LEVEL 0 TAG='FULL' DATABASE;
3. Back up the incremental level 1 copy of the entire database as a backup set.
RMAN> BACKUP INCREMENTAL LEVEL 1 TAG='INCR1' DATABASE;

Incremental restore and recovery

1. Stop the database.
\$ srvctl stop database -d mterac2
2. Restart the database in mount mode.
\$ srvctl start database -d mterac2 -o mount
3. Connect to RMAN.
\$ rman target /
4. Restore database from incremental level 0 backup.
RMAN> RESTORE DATABASE FROM TAG 'FULL';
5. Recover the database from incremental level 1 backup.
RMAN> RECOVER DATABASE FROM TAG 'INCR1';

6. Completely recover the database using archived logs and online redo log files. Make sure that the archived logs are available for database recovery.

```
RMAN> RECOVER DATABASE;
```

7. Open the database in read-write mode.

```
$ sqlplus / as sysdba
```

```
SQL> ALTER DATABASE OPEN;
```

8. Start the second database instance.

```
$ srvctl start instance -d mterac2 -i mterac22
```

Logical backup and recovery using Celerra SnapSure

The advanced backup solution leverages the NX4 advanced feature SnapSure. The Celerra SnapSure feature creates a read-only, logical point-in-time image (checkpoint) of a production file system (PFS). A logical backup is described as a virtual copy of the data files. This operation is very lightweight and gets created almost instantaneously. Depending on the state of the operating system and application, the snapshots provide a crash-consistent image of the file system. An Oracle database environment in which the database files are hosted on an NFS mounted NX4 file system can benefit from using SnapSure technology to create a secondary copy for business continuity purposes.

The following sections describe the tasks associated with backup and recovery operations.

Logical backup procedure

Creating a logical backup involves following steps:

1. Connect to the production database and place the database in archive log mode, if it is not already.

```
$ srvctl stop database -d mterac2
```

```
$ sqlplus / as sysdba
```

```
SQL> STARTUP MOUNT;
```

```
SQL> ALTER DATABASE ARCHIVELOG;
```

```
SQL> ALTER DATABASE OPEN;
```

```
$ srvctl start instance -d mterac2 -i mterac22
```

2. Place the entire database in hot backup mode.

```
SQL>ALTER DATABASE BEGIN BACKUP;
```

3. Create the read-only checkpoint of data file system /datafs on the NX4 Network Server.

```
# fs_ckpt datafs -Create -readonly y meta_savvol
```

This command creates a checkpoint and mounts /datafs_ckpt1 as a read-only file system on the same Data Mover that the datafs file system is mounted on.

4. Take the entire database out of hot backup mode.

```
SQL>ALTER DATABASE END BACKUP;
```

Logical recovery procedure

Recovering the database from logical backup up to the last committed transaction involves the following steps:

1. Connect to the production database and shut down if it is already running.

```
$ srvctl stop database -d mterac2
```

2. In current testing, as the cluster configuration files are also placed on the data file system, stop all the cluster services on both the database servers. Back up the cluster configuration files, if any changes need to be saved.

```
# crsctl stop crs
```

3. Perform a checkpoint restore of data file system /datafs on the NX4 Network Server.

```
# rootfs_ckpt datafs_ckpt1 -Restore
```

Restore the cluster configuration files from backup, if required.

As all the control file copies are available on the same file system as the data files and redo log files are placed, a consistent recoverable point-in-time copy of the database will be available, upon checkpoint restore.

4. Log in to the production database servers and verify that all the cluster services are running on both of the production database servers.

5. Connect to SQL Plus on the production database server and start up the database in mount mode.

```
$ sqlplus / as sysdba
```

```
SQL> STARTUP MOUNT;
```

6. Fully recover the database using archived logs and online redo log files.

```
SQL> ALTER DATABASE RECOVER;
```

7. Open the database in read-write mode.

```
SQL> ALTER DATABASE OPEN;
```

8. Start the second database instance.

```
$ srvctl start instance -d mterac2 -i mterac22
```

Advanced protect and recovery using Celerra Replicator (V2)

The best practice for disaster recovery of an Oracle database that uses NFS volumes is to leverage the Celerra Replicator (V2). Celerra Replicator enables you to create and manage replication sessions that produce a read-only, point-in-time copy of a given source file system at a remote destination. The Replicator (V2) sessions can be created and managed using either a powerful command line interface (CLI) or Celerra Manager. The Celerra Replicator incrementally and asynchronously transfers the data over high-speed WAN (or LAN).

The Celerra Replicator V2 feature can be leveraged to protect the Celerra file systems that host data files, control files, and archived log files in order to recover the database after a planned or unplanned shutdown of the production Celerra Network Server.

The high-level procedure for setting up remote replication of database file systems, along with performing restore and recovery at the remote site, is discussed in following sections.

Setting up communication between NX4 Network Servers at production and remote sites

This involves the following steps:

1. Set up communication on the production NX4 Network Server.

```
#nas_cel -create remote_celerra -ip 10.6.120.197 -passphrase password
```

2. Set up communication on the remote NX4 Network Server.

```
#nas_cel -create source_celerra -ip 10.6.120.177 -passphrase password
```

Setting up communication between Data Movers at the production and remote sites

This involves the following steps:

1. Set the source to target side interconnect on the production NX4 Network Server.

```
# nas_cel -interconnect -create source_remote -source_server server_2 -
destination_system remote -destination_server server_3 -source_interfaces
if3rtp118s2 -destination_interfaces if3rtp98s3
```

2. Set the target to source side interconnect on the remote NX4 Network Server.

```
# nas_cel -interconnect -create remote_source -source_server server_3 -
destination_system source -destination_server server_2 -source_interfaces
if3rtp98s3 -destination_interfaces if3rtp118s2
```

Dell recommends placing the NX4 file system replication load on the Data Mover interfaces other than the ones used to serve the production file systems to database servers.

Creating file system replication sessions between the production and remote sites

This involves the following steps:

1. Create a /datafs file system replication session from production to remote.

```
# nas_replicate -create datafs_replica -source -fs datafs -destination -fs
datafs_rep -interconnect source_remote -overwrite_destination
```

2. Create a /archfs file system replication session from production to remote.

```
# nas_replicate -create archfs_replica -source -fs archfs -destination -fs
archfs_rep -interconnect source_remote -overwrite_destination
```

Note: For these commands to be successful, the remote NX4 file systems `datafs_rep` and `archfs_rep` are expected to be available on the remote NX4 and should be of the same size as the respective production file systems. Otherwise, the `nas_replicate` command can be used to create the remote NX4 file systems by giving the `-pool` option instead of `-fs` option in the listed commands above. The `nas_replicate` command mounts the remote NX4 file systems as read-only.

Fail over to the remote site

With the file system replication session running between the production and remote sites, it is possible to perform switch-over or failover action. In the former case, it is assumed that both the NX4 Network Servers can be contacted and after performing the switch-over action, the production file system mount becomes read-only and remote file system mount becomes read-write. Therefore, care must be taken to shut down the production database before performing the switch-over action. The failover case assumes that the production NX4 Network Server is not in service, and hence a failover operation needs to be executed from a remote NX4 Network Server.

In the test, an environment failover operation was performed to validate the recovery procedure and consistency of data. The following assumption was made on the remote site:

1. Oracle two-node RAC 10g on RHEL4 (x86-64) was installed on remote database servers. The cluster configuration files path on the remote database servers is different from the path of those on primary database servers.
2. The NX4 file system /flashfs was created with the appropriate size on the respective disk drives.

Note: This file system was not part of replication.

3. The /flashfs file system has been mounted and exported on the Data Mover of the remote NX4.
4. Oracle Listener configuration has been done.

In the test environment, the production file systems were forced to fail over to the remote site.

```
# nas_replicate -failover datafs_replica
# nas_replicate -failover archfs_replica
```

Upon successful completion of this command, the /datafs_rep and /archfs_rep file systems become read-write mounted on the remote NX4 Network Server.

Recovering the database at the remote site

Recovering the database involves the following steps:

1. Export the file systems on the remote NX4 Network Server.

```
#server_export server_3 -Protocol nfs -name datafs_rep /datafs_rep
#server_export server_3 -Protocol nfs -name archfs_rep /archfs_rep
```

2. Mount the failed over NX4 file systems, datafs_rep and archfs_rep, on the database servers at the remote site.
3. As all control file copies are available on the same file system as the data files and redo log files are placed, a consistent recoverable point-in-time copy of the database is available at the remote site.
4. Create required database dump directories on remote database servers.

```
$ mkdir -p $ORACLE_HOME/admin/mterac2/
$ mkdir $ORACLE_HOME/admin/mterac2/adump
$ mkdir $ORACLE_HOME/admin/mterac2/bdump
$ mkdir $ORACLE_HOME/admin/mterac2/cdump
$ mkdir $ORACLE_HOME/admin/mterac2/ddump
$ mkdir $ORACLE_HOME/admin/mterac2/dpdump
$ mkdir $ORACLE_HOME/admin/mterac2/pfile
$ mkdir $ORACLE_HOME/admin/mterac2/udump
```

5. A copy of the production database server parameter file is available in the same path as the data files. Modify the parameter file as appropriate, name it as init{\$ORACLE_SID}.ora and place in the \$ORACLE_HOME/dbs directory, on both the database servers. Any changes to the data file path must be updated manually with the control file before opening the database.
6. Set the ORACLE_SID environmental parameter on both database servers.
7. Connect to the database on the first database server and start up with the MOUNT option.

```
$ sqlplus / as sysdba
SQL> STARTUP MOUNT;
```

10. Recover the database from archived logs as well as the online available redo logs.

```
SQL> SET AUTORECOVERY ON;
SQL> ALTER DATABASE RECOVER;
```

12. Finally, open the database.

```
SQL> ALTER DATABASE OPEN;
```

14. Register the database and both the database instances with srvctl on the first database server.

```
$ srvctl add database -d mterac2 -o $ORACLE_HOME
$ srvctl add instance -d mterac2 -i mterac21 -n node1
$ srvctl add instance -d mterac2 -i mterac22 -n node2
```

15. Start the database instance on the second database server.

```
$ srvctl start instance -d mterac2 -i mterac22
```

Restart file system replication sessions from the remote to production site

When the production site is available, restart the replication sessions from the remote site to the production site by overwriting the production file system's data.

```
# nas_replicate -start datafs_replica -reverse -overwrite_destination
```

```
# nas_replicate -start archfs_replica -reverse -overwrite_destination
```

After synchronization from the above `nas_replicate` commands, the modified data blocks will be flowing from the remote site to the production site.

Fail back to the production site

Gracefully switch back to the production site by shutting down the database on the remote site.

```
$ srvctl stop database -d mterac2
```

Reverse file system replication direction from the production site to the remote site using the following commands:

```
#nas_replicate -reverse datafs_replica
```

```
#nas_replicate -reverse archfs_replica
```

After the successful completion of the previous commands, the production file systems will be mounted read-write and the remote file systems will be mounted read-only. The file system replication data will be flowing from the production NX4 to the remote NX4.

Verify that all the cluster services are running on both the production database servers and start the database.

```
$ srvctl start database -d mterac2
```

Virtual test/dev solution using Celerra SnapSure

The Celerra SnapSure writeable checkpoints feature can be used to create a single instance test/dev database copy on a virtual machine, for Oracle RAC 10g database with the database files placed on NX4 file systems. The checkpoints will be created by placing the database in hot backup mode, thereby avoiding database downtime and with minimal overhead. The details of this solution are as follows:

In the test environment, the following assumptions were made:

- ◆ VMware ESX Server 3.5 was installed and configured on a supported hardware platform.
- ◆ A virtual machine (VM) is created with appropriate memory as required to install Oracle 10g database software.
- ◆ Red Hat Enterprise Linux 4 Update 5 is installed on the VM.
- ◆ Oracle 10g R2 database software is installed on the VM.
- ◆ Oracle Listener configuration has been done.

The following is the high-level overview of steps to be followed towards this solution:

1. Connect to the production database and place the database in archive log mode, if it is not already.

```
$ srvctl stop database -d mterac2
$ sqlplus / as sysdba
SQL> STARTUP MOUNT;
SQL> ALTER DATABASE ARCHIVELOG;
SQL> ALTER DATABASE OPEN;
$ srvctl start instance -d mterac2 -i mterac22
```

2. Place the entire database in hot backup mode.

```
SQL>ALTER DATABASE BEGIN BACKUP;
```

3. Create the writeable checkpoint of data file system /datafs on the NX4 Network Server.

```
# fs_ckpt datafs -Create -readonly y data_savvol
# fs_ckpt datafs_ckpt1 -Create -readonly n
```

The previous command creates a checkpoint and mounts /datafs_ckpt1_writeable1 as a read-write file system on the same Data Mover where the datafs file system is mounted.

4. Take the entire database out of hot backup mode.

```
SQL>ALTER DATABASE END BACKUP;
```

5. Similarly, create the writeable checkpoint of archived log file system /archfs on the NX4 Network Server.

```
# fs_ckpt archfs -Create -readonly y meta_savvol_arch
# fs_ckpt archfs_ckpt1 -Create -readonly n
```

6. Export the created writeable checkpoints.

```
#server_export server_2 -Protocol nfs -name datafs_ckpt1_writeable1
/datafs_ckpt1_writeable1
#server_export server_2 -Protocol nfs -name archfs_ckpt1_writeable1
/archfs_ckpt1_writeable1
```

7. Mount the writeable checkpoints to appropriate locations on the test/dev virtual machine. All copies of the database control file are available in the mounted dataafs writeable checkpoint along with the database data files, redo logs files, and database server parameter file.

8. Create required database dump directories on the virtual machine.

```
$ mkdir -p $ORACLE_HOME/admin/mterac2/  
$ mkdir $ORACLE_HOME/admin/mterac2/adump  
$ mkdir $ORACLE_HOME/admin/mterac2/bdump  
$ mkdir $ORACLE_HOME/admin/mterac2/cdump  
$ mkdir $ORACLE_HOME/admin/mterac2/ddump  
$ mkdir $ORACLE_HOME/admin/mterac2/dpdump  
$ mkdir $ORACLE_HOME/admin/mterac2/pfile  
$ mkdir $ORACLE_HOME/admin/mterac2/udump
```

10. Modify the parameter file as appropriate, name it as init{\$ORACLE_SID}.ora, and place in the \$ORACLE_HOME/dbs directory. Any changes to the data file path must be updated manually with the control file before opening the database.

11. Set the ORACLE_SID environmental parameter on the database server.

12. Connect to the database and start up with the MOUNT option.

```
$ sqlplus / as sysdba  
SQL> STARTUP MOUNT;
```

15. Recover the database from archived logs as well as the available online redo logs.

```
SQL> SET AUTORECOVERY ON;  
SQL> ALTER DATABASE RECOVER;
```

18. Finally, open the database.

```
SQL> ALTER DATABASE OPEN;
```


Chapter 6 Conclusion

This chapter presents the following topic:

Conclusion..... 56

Conclusion

The Dell solution for Oracle RAC 10g on Linux NFS over NX4 offers the ability to achieve a high level of scalability and performance for an entry-level solution. Combined with the high-availability features of the NX4 Network Server, the customer can implement a production Oracle database system that has enterprise-class features and benefits including:

- ◆ Data Mover failover
- ◆ Network redundancy and failover
- ◆ Network load balancing
- ◆ RAID-protected storage
- ◆ Multi-protocol SAN, iSCSI, and NAS connectivity options
- ◆ Full and incremental physical backup and recovery with RMAN
- ◆ Full and incremental logical backup and recovery with Celerra SnapSure, integrated with RMAN
- ◆ Remote replication and disaster recovery with Celerra Replicator V2
- ◆ Virtualized test/dev solution with Celerra SnapSure writeable checkpoints
- ◆ Efficient and cost-effective use of database server hardware
- ◆ Offloading backup and disaster recovery operations to storage array
- ◆ Freeing up CPU cycles for data processing
- ◆ Simplified system management
- ◆

Appendix A Sample ks.cfg

This appendix presents the following topic:

Sample ks.cfg 58

Sample ks.cfg

```

install

nfs --server=128.222.1.24 --
dir=solutions1/lab/software/Linux/Red_Hat_Enterprise_Linux/AS_4_update_5_-_AMD64-
IntelEM64T

lang en_US.UTF-8

langsupport --default=en_US.UTF-8 en_US.UTF-8

keyboard us

xconfig --card "ATI Radeon 7000" --videoram 8192 --hsync 31.5-37.9 --vsync 50-70 --
resolution 800x600 --depth 16 --startxonboot --defaultdesktop gnome

network --device eth0 --onboot yes --bootproto dhcp
network --device eth1 --onboot no --bootproto none
network --device eth2 --onboot no --bootproto none
network --device eth3 --onboot no --bootproto none
network --device eth4 --onboot no --bootproto none
network --device eth5 --onboot no --bootproto none

rootpw --iscrypted $1$rP2mLD4F$xqJrp/LiSMqOH8HVA1Xg4.

firewall --disabled

selinux --enforcing

authconfig --enablesshadow --enablemd5

timezone America/New_York

bootloader --location=mbr --append="rhgb quiet"

clearpart --all --drives=sda,sdb

part / --fstype ext3 --size=100 --grow --ondisk=sda --asprimary

part swap --size=9216 --ondisk=sdb --asprimary

%packages
@ compat-arch-development
@ admin-tools
@ editors
@ system-tools
@ text-internet
@ x-software-development
@ legacy-network-server
@ gnome-desktop
@ compat-arch-support
@ legacy-software-development
@ base-x

```

ORACLE 10G AND LINUX NFS ON DELL NX4

```
@ server-cfg
@ development-tools
@ graphical-internet
e2fsprogs
sysstat
kernel-smp-devel
kernel-devel
vnc
telnet-server
rdesktop
kernel-smp
tsclient

%post
```