White Paper

EMC PERFORMANCE FOR ORACLE

EMC VNX, Enterprise Flash Drives, FAST Cache, VMware vSphere

- Improved performance
- Easy to configure and monitor
- Scalable and nondisruptive

EMC Solutions Group

Abstract

This white paper describes the benefits of using EMC® FAST Cache for Oracle OLTP databases in both physical and virtual environments. The Oracle RAC 11*g* database was configured to access EMC VNX7500™ file storage over NFS, using the Oracle dNFS Client. VMware® vSphere™ provided the virtualization platform for the virtual environment.

December 2011

Part Number H8850.4

Table of contents

Executive summary 5
- Business case 5
- Solution overview 5
- Key results 6

Introduction 7
- Purpose 7
- Scope 7
- Audience 7
- Terminology 7

Key technology components 9
- Overview 9
- EMC VNX7500 9
- EMC FAST Cache 9
  - Overview 9
  - How FAST Cache works 10
  - FAST Cache features 11
- VMware vSphere 11
- Oracle 12

Solution architecture and design 13
- Solution overview 13
- Architecture of physical environment 13
- Architecture of virtual environment 13
- Hardware resources 14
- Software resources 15
- Oracle storage layout 15
- Oracle file system allocation on the VNX7500 16
- Oracle file system allocation on host 16
- Oracle dNFS Client configuration 16
- Physical server configuration 17

Oracle Database configuration 19
- Database and workload profile 19
- Database schema 19

VMware ESX server configuration 21
- Overview 21
- Step 1: Create the virtual switches 21

Step 2: Configure the virtual machine template ... 22
Step 3: Deploy the virtual machines ... 23
Step 4: Enable access to the storage devices ... 24

Configuring FAST Cache on the VNX ... 25
Overview ... 25
Prerequisites for using FAST Cache ... 25
Analyze the application workload ... 25
Prerequisites for creating FAST Cache ... 26
Create the FAST Cache ... 26
Enable or disable FAST Cache ... 28
FAST Cache best practices for Oracle ... 28

Live migration from physical to virtualized Oracle RAC ... 29
Overview ... 29
Live migration process ... 29
Configure SSH user equivalence ... 30
Configure the private network interconnect ... 30
Adding the virtual machines as cluster nodes ... 31
Adding Oracle RAC database instances ... 31
Deleting the Oracle RAC physical nodes ... 31

Testing and validation ... 33
FAST Cache and manual tiering comparison ... 33
FAST Cache test procedure ... 33
FAST Cache read/write hit ratio ... 34
Wait statistics from Oracle AWR reports ... 35
FAST Cache effects on transactions per minute ... 36
FAST Cache effects on response times ... 38
FAST Cache effects on IOPS ... 39

Conclusion ... 40
Summary ... 40
Findings ... 40

References ... 41
White papers ... 41
Product documentation ... 41
Other documentation ... 41

Executive summary

Business case

Oracle Database OLTP applications tend to be mission-critical, with low I/O latency requirements. Enterprise Flash drives can dramatically increase performance for these applications.

However, to get the maximum benefit from Flash drives, without increasing total cost of ownership (TCO), administrators must continually identify and place only the most active data on these devices. Traditionally, this was a complex, repetitive, manual process, which involved disruption to the normal operation of the database. Moving data between storage tiers either required the database to be taken down or caused considerable performance penalties if performed online.

By deploying the new EMC® VNX™ family of unified storage platforms, in conjunction with EMC FAST Cache technology, enterprises can circumvent the performance and operational issues that arise as Oracle OLTP databases outgrow the available capacity on the highest performing drive types.

FAST Cache is a storage performance optimization feature that provides immediate access to frequently accessed data. It uses dedicated Flash drives to provide an extra layer of cache for the most active data and automatically moves data in and out of this new cache based on usage patterns.

This reduces the need to add more Flash drives to maintain or increase performance. A small number of Flash drives implemented as FAST Cache provides a greater performance increase than the more traditional method of increasing the number of hard disk drives (HDDs) and spreading the application load across these drives (also known as short stroking).

In addition, in virtual environments, FAST Cache can have the practical effect of eliminating the small I/O overhead that may be introduced by virtualization.

Solution overview

This solution demonstrates how FAST Cache technology can be leveraged by Oracle OLTP applications to improve performance in both physical and virtual environments on EMC VNX storage. Testing was carried out on an Oracle Real Application Clusters (RAC) 11*g* database that was configured to access the VNX7500™ file storage over Network File System (NFS), using the Oracle Direct NFS (dNFS) Client.

The solution also includes live migration from the physical environment to the virtual environment, which is provided by VMware® vSphere™.

Key results

The key benefits demonstrated by the solution include:

- **Improved performance**
  - Over 100 percent improvement in transactions per minute (TPM)
  - A 170 percent improvement in I/Os per second (IOPS)
  - Over 79 percent decrease in average transaction response times
- **Automatic, nondisruptive operation**

  Data is cached in and out of FAST Cache automatically and nondisruptively.
- **Scalable capacity**

  FAST Cache can be easily and nondisruptively extended as application needs evolve, enabling customers to take an incremental approach to Flash drive deployment.
- **Excellent fit for Oracle OLTP databases**

  FAST Cache is particularly suited to database applications that generate a large number of random I/Os, that experience sudden bursts in user query activity or high, concurrent user loads, and where the working data set fits within the Flash drive cache.
- **Easy to configure and monitor**

  FAST Cache can be configured with a few simple steps and can be enabled or disabled for individual logical unit numbers (LUNs) with a single click.

Introduction

Purpose

This white paper introduces EMC FAST Cache technology and how it can be leveraged by Oracle OLTP applications to improve performance in both physical and virtual environments on EMC VNX storage.

Scope

The scope of this white paper is to:

- Present an overview of the key technology components involved in the solution.
- Document details of the architecture and design of the solution infrastructure in both the physical and virtual environments.
- Describe how Oracle and FAST Cache were configured for the solution.
- Describe the steps used to migrate the database from the physical environment to the virtual environment.
- Present the results of the tests performed and the positive effects of using FAST Cache.
- Identify the key business benefits of using FAST Cache in Oracle RAC environments.

Audience

This white paper is intended for database administrators, storage administrators, VMware administrators, EMC customers, and field personnel who want to improve the performance of business applications by implementing FAST Cache technology in their Oracle environments using EMC VNX unified storage. It is assumed that the reader is familiar with the following products:

- EMC VNX storage
- VMware vSphere and VMware templates
- Oracle Database 11*g* R2 Enterprise Edition, Oracle Grid Infrastructure, and Oracle RAC

Terminology

This paper includes the following acronyms

Table 1. Acronyms

Acronym	Term
AWR	Automatic Workload Repository
CIFS	Common Internet File System
DBCA	Database Configuration Assistant
dNFS	Direct NFS
FAST VP	Fully Automated Storage Tiering for Virtual Pools
FC	Fibre Channel
FCoE	Fibre Channel over Ethernet

Acronym	Term
HAIP	Highly Available IP
HDD	Hard disk drive
IOPS	I/Os per second
iSCSI	Internet SCSI
LUN	Logical unit number
NIC	Network interface card
NFS	Network File System
ODM	Oracle Disk Manager
OLTP	Online transaction processing
RAC	Real Application Clusters
RHEL	Red Hat Enterprise Linux
SAS	Serial Attached SCSI
SCSI	Small Computer System Interface
SGA	System global area
SSH	Secure Shell
TCO	Total cost of ownership
TPM	Transactions per minute
vDS	vNetwork Distributed Switch
VNX OE	VNX Operating Environment

Key technology components

Overview

The solution uses the following hardware and software components:

- EMC VNX7500
- EMC FAST Cache
- VMware vSphere
- Oracle Database 11*g* Release 2 Enterprise Edition with Oracle Clusterware

EMC VNX7500

For the solution, storage is provided by an EMC VNX7500 storage array. The VNX7500 is a member of the VNX series next-generation storage platform, which is powered by Intel quad-core Xeon 5600 series processors and delivers five 9's availability. The VNX series is designed to deliver maximum performance and scalability for enterprises, enabling them to dramatically grow, share, and cost-effectively manage multi-protocol file and block systems.

The VNX Operating Environment (VNX OE) allows Microsoft Windows and Linux/UNIX clients to share files in multi-protocol NFS and Common Internet File System (CIFS) environments. At the same time, VNX OE supports Internet SCSI (iSCSI), Fibre Channel (FC), and Fibre Channel over Ethernet (FCoE) access for high-bandwidth and latency-sensitive block applications.

EMC VNX protects against hardware or software failure by providing one or more standby Data Movers, and for the VNX7500, a standby Control Station. The standby Data Mover or Control Station assumes operation from the failed component. For further information, refer to the EMC technical module *Configuring Standbys on VNX*.

EMC Unisphere™ is the central management platform for the VNX series, providing a single, combined view of file and block systems, with all features and functions available through a common interface. Unisphere is optimized for virtual applications and provides industry-leading VMware integration, automatically discovering virtual machines and ESX servers and providing end-to-end, virtual-to-physical mapping. Unisphere also simplifies the configuration of FAST Cache on VNX platforms.

EMC FAST Cache

Overview

FAST Cache is part of the FAST Suite for VNX arrays, which also includes Fully Automated Storage Tiering for Virtual Pools (FAST VP).

FAST VP automatically moves data to the most appropriate storage tier based on sustained data access and demands over time. FAST Cache automatically absorbs unexpected spikes in application workloads, providing immediate performance benefits for burst-prone data.

FAST Cache and FAST VP can be used alone or together (see Figure 1). This solution demonstrates the benefits of FAST Cache only.

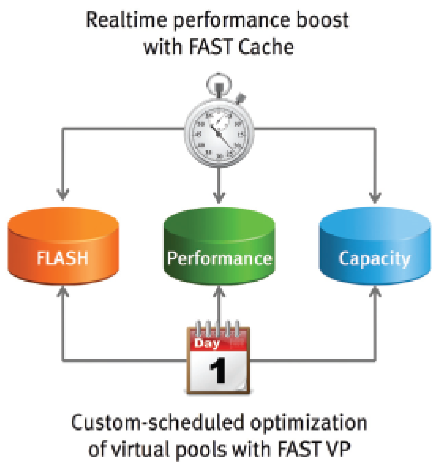


Figure 1. FAST Suite

FAST Cache uses Flash drives to add an extra layer of cache between DRAM cache and rotating disk drives, thereby creating a faster medium for storing frequently accessed data. FAST Cache is an extendable, read/write cache. It boosts application performance by ensuring that the most active data is served from high-performing Flash drives and can reside on this faster medium for as long as is needed.

FAST Cache tracks data activity at a granularity of 64 KB and promotes hot data into FAST Cache by copying it from the HDDs to the Flash drives assigned to FAST Cache. Subsequent I/O access to that data is handled by the Flash drives and is serviced at Flash drive response times—this ensures very low latency for the data. As data ages and becomes less active, it is flushed from FAST Cache to be replaced by more active data.

A small number of Flash drives implemented as FAST Cache provides a greater performance increase than a large number of short-stroked HDDs.

FAST Cache is particularly suited to applications that randomly access storage with high frequency, such as Oracle OLTP databases. In addition, OLTP databases have inherent locality of reference with varied I/O patterns. Applications with these characteristics benefit most from deploying FAST Cache. The optimal use of FAST Cache is achieved when the working data set can fit within the FAST Cache.

How FAST Cache works

Figure 2 Illustrates how FAST Cache works.

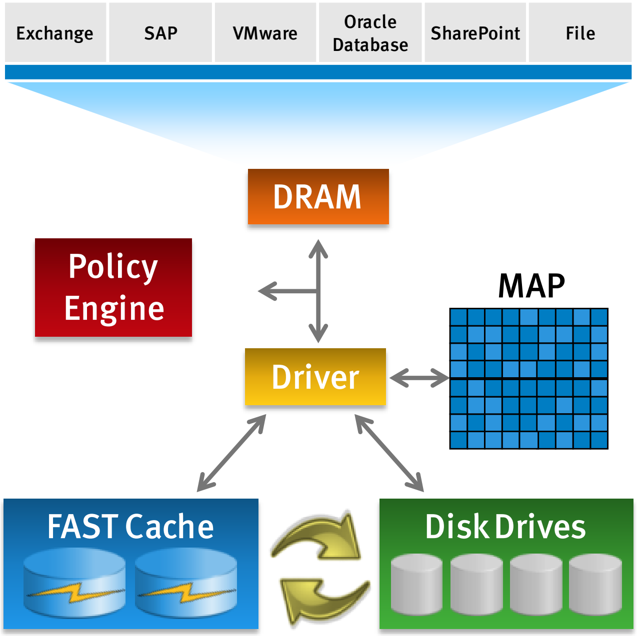


Figure 2. How FAST Cache works

Requests

- Requests for data are serviced from DRAM if the data is cached there.
- If not, the FAST Cache driver checks the FAST Cache memory map. If the data is in FAST Cache, then the request is serviced from there.

  Note: The FAST Cache memory map tracks and maintains information about each 64 KB chunk of storage that resides in FAST Cache.

- If the page is not in FAST Cache, the request is serviced from disk.

Promotion/demotion

- The FAST Cache Policy Engine manages the flow of I/O through FAST Cache.
- If data is being used frequently, the Policy Engine promotes it to FAST Cache and subsequent requests for the data are serviced from there. As the data becomes less active, it is copied back to the disk drives as background activity.
- The Policy Engine determines when these actions should take place.

A write operation works in a similar way. Writes with high locality of reference are directed to the Flash drives. When the time comes to flush this data to disk, the flushing operation is significantly faster as writes are now at Flash drive speeds.

FAST Cache features

FAST Cache includes the following features:

- It tracks data temperature at a granularity of 64 KB, continuously ensuring that the hottest data is served from high-performance Flash drives.
- It promotes and flushes out data automatically and transparently, depending on data usage patterns.
- It is read/write in nature, so it also accelerates data write rehits.
- It can be enabled and disabled at the storage pool or LUN level.
- It supports both file and block on VNX arrays.
- It can be managed through Unisphere in an easy and intuitive manner.
- Flash drives can be added to FAST Cache without incurring array downtime.

VMware vSphere

VMware vSphere provides the virtualization platform for the solution, with VMware ESX® virtual machines hosting the Oracle RAC nodes in the virtual environment.

VMware vSphere abstracts applications and information from the complexity of the underlying infrastructure, through comprehensive virtualization of the server, storage, and networking hardware. It is the industry's most complete and robust virtualization platform, virtualizing business-critical applications with dynamic resource pools for unprecedented flexibility and reliability.

VMware vCenter™ provides the centralized management platform for vSphere environments, enabling control and visibility at every level of the virtual infrastructure.

Oracle

This solution demonstrates the performance benefits of using FAST Cache for consolidated Oracle OLTP database environments. It takes advantage of Oracle Database 11*g* R2 features such as Oracle RAC and Oracle dNFS:

- Oracle RAC extends Oracle Database so that you can store, update, and efficiently retrieve data using multiple database instances on different servers at the same time. Oracle RAC provides the software that manages multiple servers and instances as a single group.
- Oracle dNFS Client is an optimized NFS client that is built directly into the database kernel. This native capability enables direct I/O with the storage devices, bypassing the operating system file cache and reducing the need to copy data between the operating system and database memory. The dNFS Client also enables asynchronous I/O on NFS appliances.
- For Oracle 11*g* RAC databases, dNFS typically improves database performance by 10 to 15 percent when compared to the standard kernel NFS. For more information refer to the white paper: *Optimizing EMC Celerra IP Storage on Oracle 11g Direct NFS*.

In Oracle Database 11*g* R2, Oracle Clusterware, which is the underlying clustering software required to run an Oracle RAC database, is part of the Oracle Grid Infrastructure software bundle.

Solution architecture and design

Solution overview

This EMC Oracle Performance solution is designed to test the effects of FAST Cache on the performance of an Oracle OLTP database in both physical and virtual environments. Testing was carried out on an Oracle RAC 11*g* database, with an EMC VNX7500 array providing the underlying storage, and VMware vSphere providing the virtualization platform.

VNX arrays support a variety of file access protocols. When a VNX array is configured as an NFS server, file systems are mounted on the VNX and the paths to the file systems are exported. The exported file systems are then available over IP and can be mounted by remote clients. For the solution, the Oracle RAC nodes were configured to access the NFS server directly using the Oracle internal dNFS Client.

For both physical and virtual environments, four 200 GB Flash drives were identified as the optimal configuration for FAST Cache (see Configuring FAST Cache on the VNX › Analyze the application workload) and this configuration was used for testing.

Architecture of physical environment

Figure 3 depicts the architecture of the physical environment. Two physical servers were used to deploy a two-node Oracle RAC cluster. The storage and cluster interconnect networks used 10 Gigabit Ethernet.

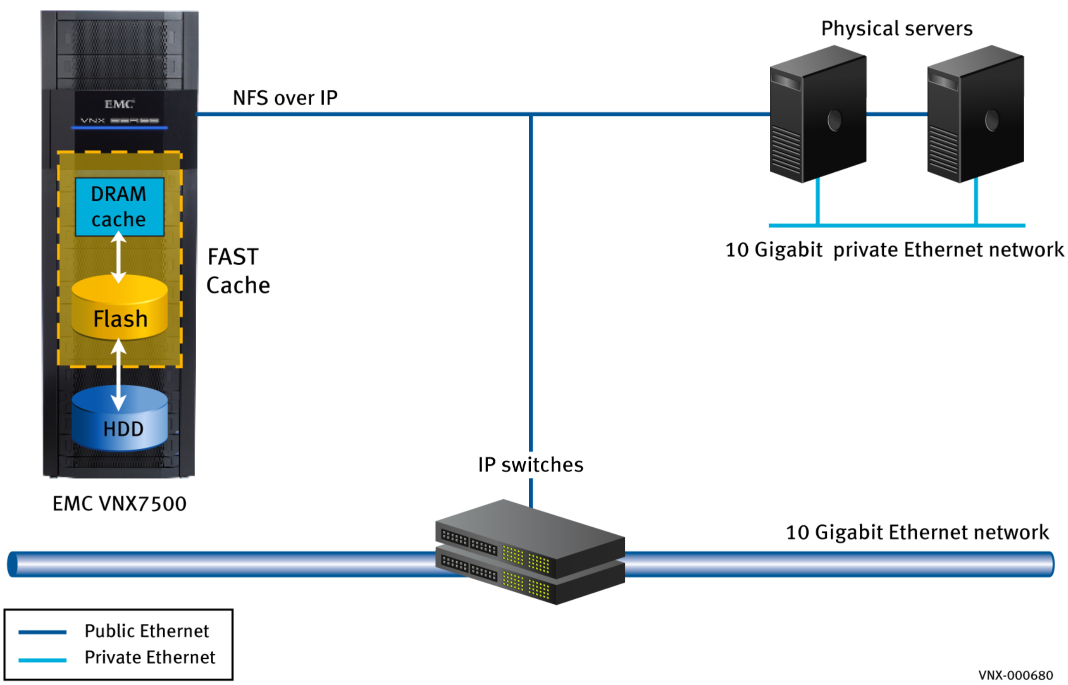


Figure 3. Solution architecture – physical environment

Architecture of virtual environment

Figure 4 depicts the architecture of the virtual solution. The VMware ESX server farm consisted of two ESX servers with ESX version 4.1 installed. Two Red Hat Enterprise Linux (RHEL) 5.5 virtual machines from each ESX server were used to form a four-node RAC cluster database. In line with Oracle's support position for Oracle products running on VMware, the solution was deployed using Oracle RAC 11.2.0.2.

The same NFS storage as the physical environment was made available as mount points to all virtual servers.

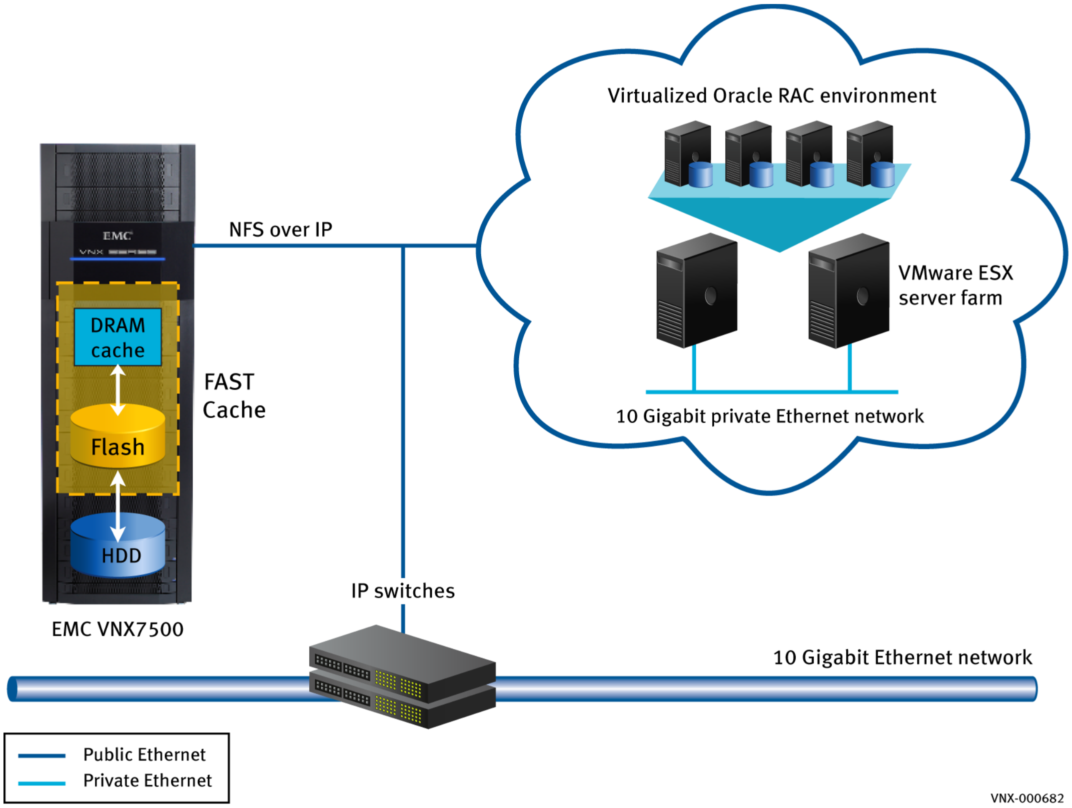


Figure 4. Solution architecture – virtual environment

Hardware resources

Table 2 details the hardware resources for the solution.

Table 2. Hardware environment

Purpose	Quantity	Configuration
Storage array	1	EMC VNX7500, with: • 62 x 600 GB 15k SAS drives • 4 x 600 GB 15k SAS drives (vault) • 4 x 200 GB Flash drives
Oracle RAC database servers (physical environment)	2	Linux server with: • 4 x eight-core CPU • 128 GB RAM • Dual 1 Gb NICs • Dual 10 Gb CNAs
VMware ESX servers (virtual environment)	2	
Ethernet switches	2	10 Gb/s Ethernet switches

Note In both physical and virtual environments, jumbo frames were configured for the VNX Data Movers, the Oracle RAC servers, and the Ethernet switches.

Software resources

Table 3 details the software resources for the solution.

Table 3. Solution software environment

Software	Version	Purpose
EMC VNX OE for block	05.31.007.3.126	VNX operating environment
EMC VNX OE for file	7.0.14.0	VNX operating environment
Unisphere	1.1.0.1.0387	VNX management software
VMware vSphere	4.1	Hypervisor hosting all virtual machines
VMware vCenter	4.1	Management of VMware vSphere
Oracle Database 11*g* R2	Enterprise Edition 11.2.0.2	Oracle database and cluster software
RHEL	5.5	Application and database server OS

Oracle storage layout

The disk configuration used four back-end 6 Gb Serial Attached SCSI (SAS) ports within the VNX 7500. Figure 5 provides a logical representation of the file system used for the Oracle data files. Four Data Movers were used in a 2+2 active/standby configuration. The two active Data Movers were used to access the file systems, which were distributed evenly across the four SAS ports. The back-end configuration was based on the I/O requirements and follows EMC best practice. Unisphere provides a simple, easy-to-use GUI to create and manage the file systems.

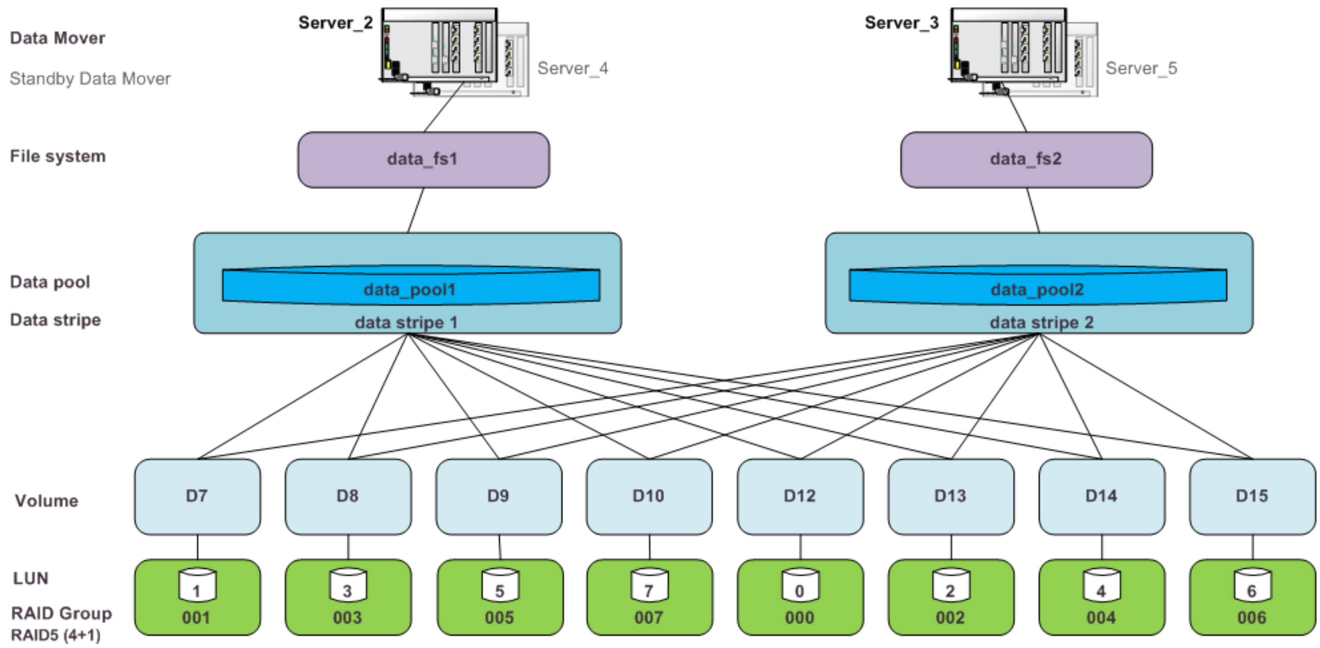


Figure 5. Data file system logical view

Oracle file system allocation on the VNX7500

Table 4 details the Oracle file system storage allocation on the VNX7500. All the RAID groups were created on 600 GB 15k SAS drives.

Table 4. Oracle file system allocation on the VNX7500

File type	Raid type	No. of LUNs	Disk volumes (dVols)	Size	Data Mover
FRA files	4+1 RAID 5	2	D17, D19	4 TB	Server2
CRS files	2+2 RAID 10	1	D18	10 GB	Server2
Data files, temp files, control files	4+1 RAID 5	8	D7 to D10, D12 to D15	2 TB	Server2
	4+1 RAID 5	8	D7 to D10, D12 to D15	2 TB	Server3
Redo logs	2+2 RAID 10	1	D23	500 GB	Server2
	2+2 RAID 10	1	D24	500 GB	Server3

Oracle file system allocation on host

Table 5 details the NFS access points and host mount points for the Oracle file system.

Table 5. Oracle file system allocation on host

File type	Mount point	IP address
FRA files	fra_fs	192.168.4.80/81
CRS files	crs_fs	192.168.4.80/81
Data files, temp files, control files	data_fs1	192.168.4.80/81
	data_fs2	192.168.5.80/81
Redo logs	log_fs1	192.168.4.80/81
	log_fs2	192.168.5.80/81

Oracle dNFS Client configuration

Oracle dNFS Client was configured for both the physical and virtual environments. The Oracle file systems were mounted and made available over regular NFS mounts. Oracle dNFS Client then used the **oranfstab** configuration file to determine the mount point settings for the NFS storage devices. Figure 6 shows an extract from the **oranfstab** file used for the solution.

```
server: 192.168.4.81
local: 192.168.4.56
path: 192.168.4.81
export: /data_fs1 mount: /mnt/data_fs1
export: /fra_fs mount: /mnt/fra_fs
export: /log_fs1 mount: /mnt/log_fs1
export: /crs_fs mount: /mnt/crs_fs
server: 192.168.5.81
local: 192.168.5.56
path: 192.168.5.81
export: /log_fs2 mount: /mnt/log_fs2
export: /data_fs2 mount: /mnt/data_fs2
```

Figure 6. Extract from oranfstab configuration file

The standard Oracle Disk Manager (ODM) library was replaced with one that supports dNFS Client. Figure 7 shows the commands that enable the dNFS Client ODM library.

```
[oracle@ora-rh3]$ cd $ORACLE_HOME
[oracle@ora-rh3]$ cp libodm11.so libodm11.so_stub
[oracle@ora-rh3]$ ln -s libnfsodm11.so libodm11.so
```

Figure 7. Enabling the dNFS Client ODM library

Physical server configuration

Table 6 outlines the configuration of the database servers in the physical environment. Both servers had identical configurations.

Table 6. Physical server configuration

Part	Description
CPU	4 x 8-core CPUs
Memory	128 GB, with 50 GB of memory assigned to the Oracle SGA
Operating system	Red Hat Enterprise Linux Server release 5.5 (Tikanga) 64-bit
Kernel	2.6.18-194.el5
Network interfaces	Eth0: public/management IP network Eth1 (10 Gb): dedicated to cluster interconnect Eth2 (10 Gb): dedicated to NFS connection for Data Mover 2 Eth3 (10 Gb): dedicated to NFS connection for Data Mover 3
OS user	Username: oracle UserID:501
OS groups	Group: oinstall GroupID:1000 Group: dba GroupID:1200
rpm packages installed (as Oracle prerequisites)	See the relevant Oracle installation guide.
Disk configuration	67 GB – disk for root , Oracle 11*g* R2 Grid, and RAC Database binaries 63 GB – disk for /tmp **Note:** As of Oracle Grid Infrastructure 11.2.0.2, allow for an additional 1 GB of disk space per node for the Cluster Health Monitor (CHM) Repository. By default, this resides within the Grid Infrastructure home.

Part	Description
System configuration (as Oracle prerequisites)	See the relevant Oracle Installation Guide: • *Oracle Real Application Clusters Installation Guide 11g Release 2 (11.2) for Linux* • *Oracle Grid Infrastructure Installation Guide 11g Release 2 (11.2) for Linux*

Oracle Database configuration

Database and workload profile

Table 7 details the database and workload profile for the solution.

Table 7. Database and workload profile

Profile characteristic	Details
Database type	OLTP
Database size	1 TB
Oracle RAC (physical environment)	2 nodes
Oracle system global area (SGA) for physical environment	50 GB per node
Oracle RAC (virtual environment)	4 nodes
Oracle system global area (SGA) for virtual environment	24 GB per node
Workload profile	Swingbench Order Entry (TPC-C-like) workload
Database read/write ratio	60/40
User scaling	600 to 2,000, in increments of 200

Database schema

The Swingbench Order Entry - PL/SQL (SOE) schema was used to deliver the OLTP workloads required by the solution. This schema models a traditional OLTP database. Tables and indexes reside in separate tablespaces. The SOE schema used in this solution contains the tables and indexes shown in Table 8.

Table 8. Schema tables and indexes

Table name	Index
CUSTOMERS	CUSTOMERS_PK (UNIQUE), CUST_ACCOUNT_MANAGER_IX, CUST_EMAIL_IX, CUST_LNAME_IX, CUST_UPPER_NAME_IX
INVENTORIES	INVENTORY_PK (UNIQUE), INV_PRODUCT_IX, INV_WAREHOUSE_IX
ORDERS	ORDER_PK (UNIQUE), ORD_CUSTOMER_IX, ORD_ORDER_DATE_IX, ORD_SALES_REP_IX, ORD_STATUS_IX
ORDER_ITEMS	ORDER_ITEMS_PK (UNIQUE), ITEM_ORDER_IX, ITEM_PRODUCT_IX
PRODUCT_DESCRIPTIONS	PRD_DESC_PK (UNIQUE), PROD_NAME_IX
PRODUCT_INFORMATION	PRODUCT_INFORMATION_PK (UNIQUE), PROD_SUPPLIER_IX
WAREHOUSES	WAREHOUSES_PK (UNIQUE)
LOGON	–

The SOE schema was mapped to the fastdb database service. This enabled load balancing and failover across the RAC cluster in both the physical and virtual environments (see Table 9).

Table 9. Oracle RAC database configuration

Environment	Schema	Service	RAC node/instance
Physical	SOE	fastdb	ora-rh1/fastdb1 ora-rh2/fastdb2
Virtual			ora-rh3/fastdb3 ora-rh4/fastdb4 ora-rh5/fastdb5 ora-rh6/fastdb6

VMware ESX server configuration

Overview

For the virtual environment, two ESX servers were configured on the same server hardware model used in the physical environment. Two virtual machines were created on each ESX server to form a four-node Oracle RAC cluster.

The virtual machines were created using a VMware template. First, an RHEL 5.5 virtual machine was created and Oracle prerequisites and software were installed. A template of this virtual machine was then created and used to create the other virtual machines to be used as cluster nodes.

The main steps for configuring the ESX servers are:

1. Create the virtual switches for the cluster interconnects and connection to the NFS server.
2. Configure the virtual machine template.
3. Deploy the virtual machines.
4. Enable virtual machine access to the storage devices.

Step 1: Create the virtual switches

One standard vSwitch and two vNetwork Distributed Switches (vDS) were created on the ESX servers.

The standard vSwitch was a public network configured with two 1 Gb network interface cards (NICs) for fault tolerance, as shown in Figure 8.

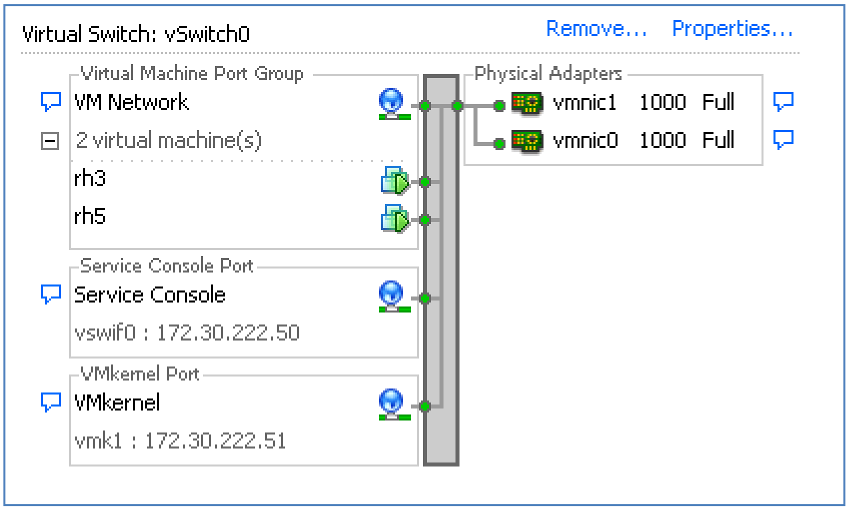


Figure 8. Standard vSwitch configuration

Each vDS was configured with next generation 10 Gb Ethernet connectivity. dvSwitch1 – Interconnect was a private network dedicated to the Oracle cluster interconnect. dvSwitch2 – Storage was a private network serving the NFS storage array.

Each switch was created with a dvPort group and an uplink port group. The uplink port group was served by two uplinks. Each uplink used a physical NIC from each ESX server for load balancing and fault tolerance, as shown in Figure 9.

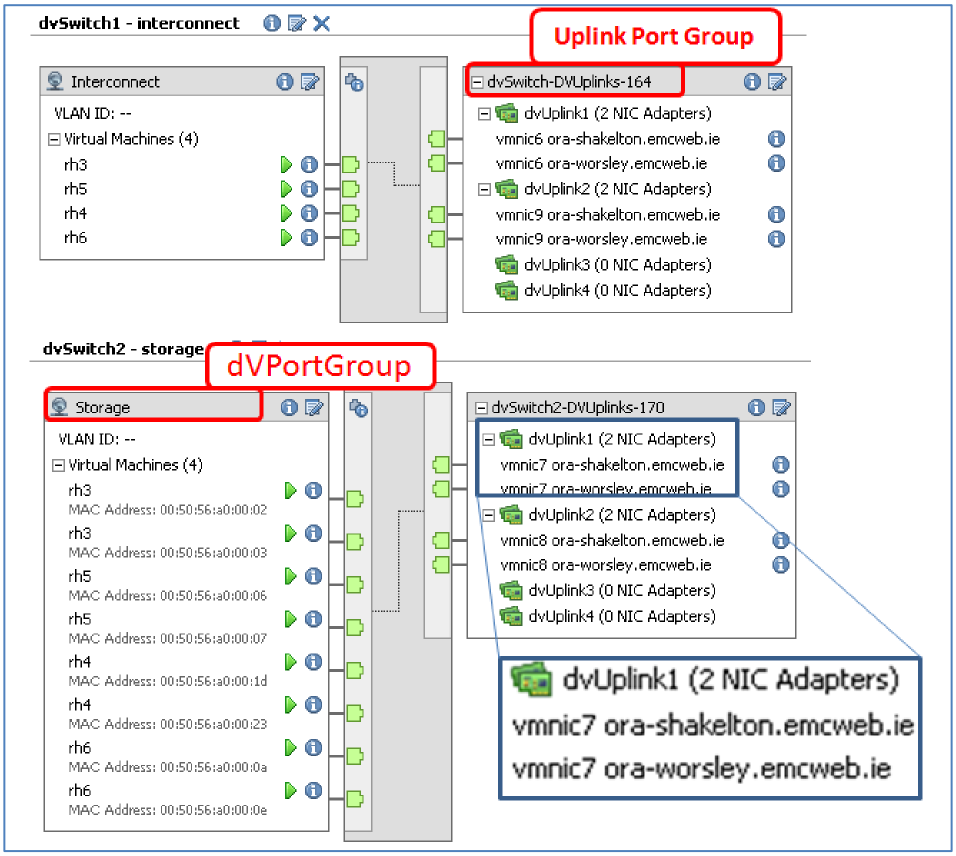


Figure 9. vNetwork Distributed Switch configuration

Note In line with *My Oracle Support Note 1212703.1*, ensure that multicasting is enabled on the switch used for the interconnect. In Oracle Grid Infrastructure 11.2.0.2, the new Redundant Interconnect Usage feature requires multicast network communication on the private interconnect network (cluster_interconnect).

Step 2: Configure the virtual machine template

The virtual machine template was configured (in VMware vSphere Client) with the requirements and prerequisites for the Oracle software (see Table 10), including:

- Operating system and rpm packages
- Kernel configuration
- OS users
- Supporting software

Table 10. Virtual machine template configuration

Part	Description
CPU	8 vCPUs
Memory	32 GB
Operating system	RHEL Server release 5.5 (Tikanga) 64-bit
Kernel	2.6.18-194.el5

Part	Description
Network interfaces	Eth0: public/management IP network Eth1 (10 Gb): dedicated to cluster interconnect Eth2 (10 Gb): dedicated to NFS connection to Data Mover 2 Eth3 (10 Gb): dedicated to NFS connection to Data Mover 3
OS user (user created and password set)	Username: oracle UserID:501
OS groups	Group: oinstall GroupID:1000 Group: dba GroupID:1200
Software pre-installed	The script sshUserSetup.sh was copied from the Oracle Grid Infrastructure 11*g* R2 binaries to /home/oracle/sshUserSetup.sh.
rpm packages installed (as Oracle prerequisites)	See the relevant Oracle installation guide.
Disk configuration	25 GB – virtual disk for root, /tmp, and the swap space 15 GB – virtual disk for Oracle 11*g* R2 Grid and RAC Database binaries **Note:** As of Oracle Grid Infrastructure 11.2.0.2, allow for an additional 1 GB of disk space per node for the Cluster Health Monitor (CHM) Repository. By default, this resides within the Grid Infrastructure home.
System configuration (as Oracle prerequisites)	See the relevant Oracle Installation Guide: *Oracle Real Application Clusters Installation Guide 11g Release 2 (11.2) for Linux* *Oracle Grid Infrastructure Installation Guide 11g Release 2 (11.2) for Linux*

Step 3: Deploy the virtual machines

Three virtual machines were deployed from the template image held in VMware vCenter. The **Deploy Template** wizard was used to specify the name and location for the new virtual machines and the customization option for the guest operating system.

A previously created customization specification (held in vCenter) defined the configuration of the network interfaces for the new virtual machines, as shown in Figure 10.

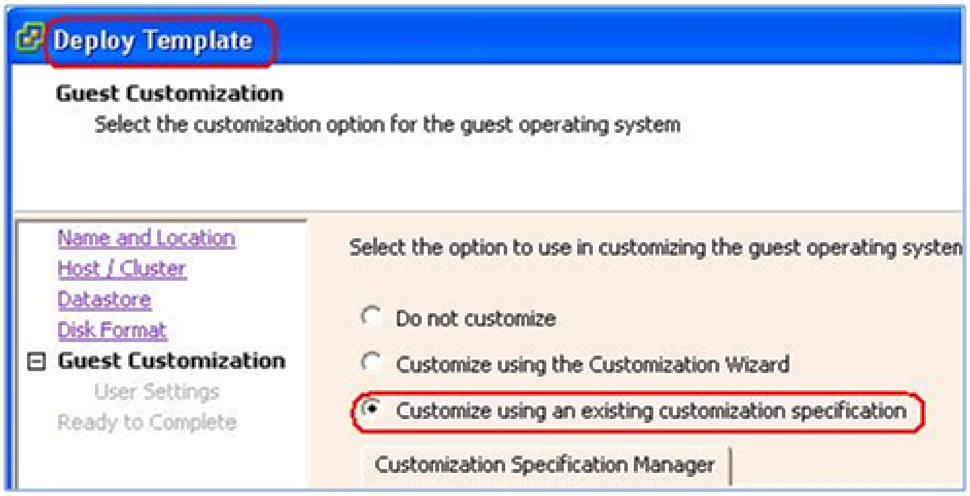


Figure 10. Deploy Template wizard

Step 4: Enable access to the storage devices

To enable host access using the Unisphere GUI, use the **Create NFS Export** option under **Storage › Shared folder › NFS**, and enter the host IP addresses for each NFS export, as shown in Figure 11.

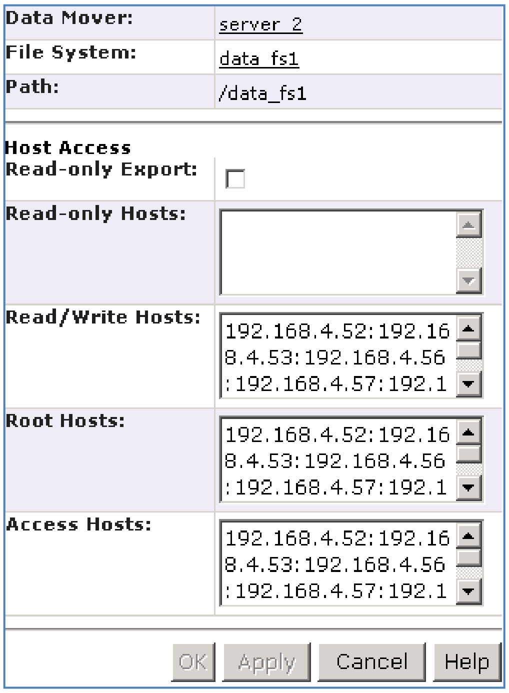


Figure 11. Configure host access

Configuring FAST Cache on the VNX

Overview

FAST Cache provides read/write caching using a private RAID 1 LUN consisting of Flash disks. A LUN on a RAID group without Flash disks can use the FAST Cache if FAST Cache is enabled for the LUN.

Prerequisites for using FAST Cache

- The application workload **must first be analyzed** to determine whether the application will benefit from FAST Cache and to determine the optimal size of the FAST Cache.
- The storage system must have the FAST Cache enabler installed.
- The storage system must have Flash disks that are not already in a storage pool.
- FAST Cache must be configured on the storage system.
- FAST Cache must be enabled for the RAID group LUNs and/or the storage pools that are to use the FAST Cache.

This section discusses the main prerequisites for installing and using FAST Cache and outlines the main steps carried out to configure and enable FAST Cache for the solution. The configuration steps can be carried out using either the Unisphere GUI or the Unisphere CLI.

For further information on configuring FAST Cache, consult *Unisphere Help* in the Unisphere GUI.

Analyze the application workload

The decision to implement FAST Cache should only be made after the application workload characteristics are measured and analyzed. Array-level tools are available to EMC field and support personnel for determining both the suitability of FAST Cache for a particular environment and the right size FAST Cache to configure. Contact your EMC sales teams for guidance.

Whether a particular application will benefit from using FAST Cache, and what the optimal cache size should be, is determined by the size of the application's active working set, the IOPS requirement, the RAID type, and the read/write ratio. As indicated in the Key technology components › EMC FAST Cache section of this white paper, the workload characteristics of OLTP databases make them especially suitable for using FAST Cache. For further information, refer to the white papers: *EMC CLARiiON, Celerra Unified, and VNX FAST Cache—A Detailed Review* and *Deploying Oracle Database on EMC VNX Unified Storage.*

For the solution, the analysis performed using the EMC array-level tools confirmed that the Oracle database workload would benefit from using FAST Cache and indicated that four 200 GB Flash drives would be the optimal configuration.

In line with EMC recommendations, FAST Cache was enabled for the Oracle data files only. Oracle archive files and redo log files have a predictable workload composed mainly of sequential writes. These can be efficiently handled by the array's write cache and assigned HDDs. Enabling FAST Cache on these files is neither beneficial nor cost effective.

Prerequisites for creating FAST Cache

Before creating FAST Cache, the following requirements must be met:

- The FAST Cache enabler must be installed on the VNX array.

  To check this, open the **Software** tab in the **System Properties** dialog box in Unisphere (see Figure 12).

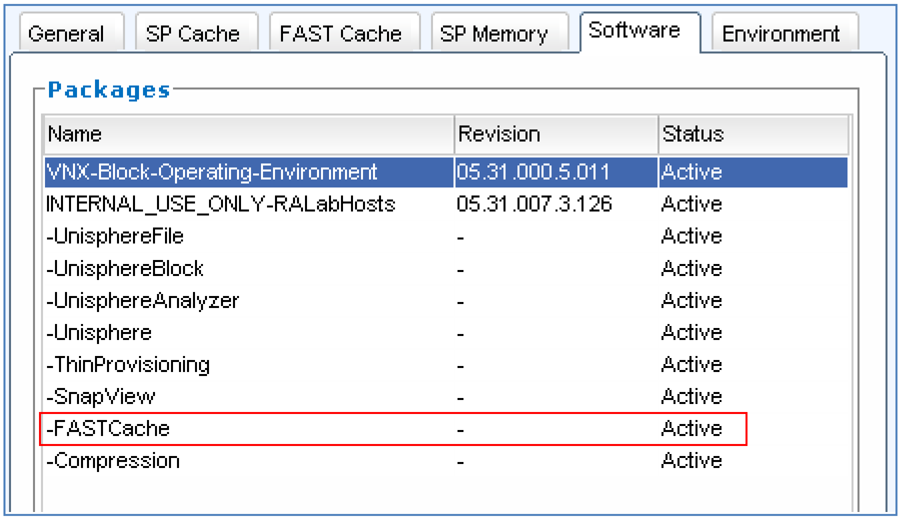


Figure 12. Confirming that FAST Cache is installed

- The storage system must have Flash disks that are not already bound to a storage pool, as a Flash disk cannot be used for both storage and cache at the same time.

  To check this, open **Disks** view in Unisphere and ensure that there are sufficient unbound Flash disks to create FAST Cache of the required size.

Create the FAST Cache

To create FAST Cache using the Unisphere GUI, access the **FAST Cache** tab under **System Management › Manage Cache** and click **Create**.

In the **Create FAST Cache** dialog box, specify the FAST Cache drive criteria (RAID type and number of disks) and select the disks to be used for FAST Cache. The **Disks** section lists the disks available for FAST Cache and their properties—the system can select the FAST Cache disks for you, based on the specified drive criteria, or you can select them manually.

Figure 13 shows the FAST Cache configuration for the solution:

- **Number of Disks** is set to 4, as recommended by the workload analysis.
- **RAID Type** is set to RAID 1, as this is currently the only RAID type supported.
- Four disks were manually selected from the list of available disks, spread over two buses (Bus 2 and Bus 0). This ensures a balanced workload.

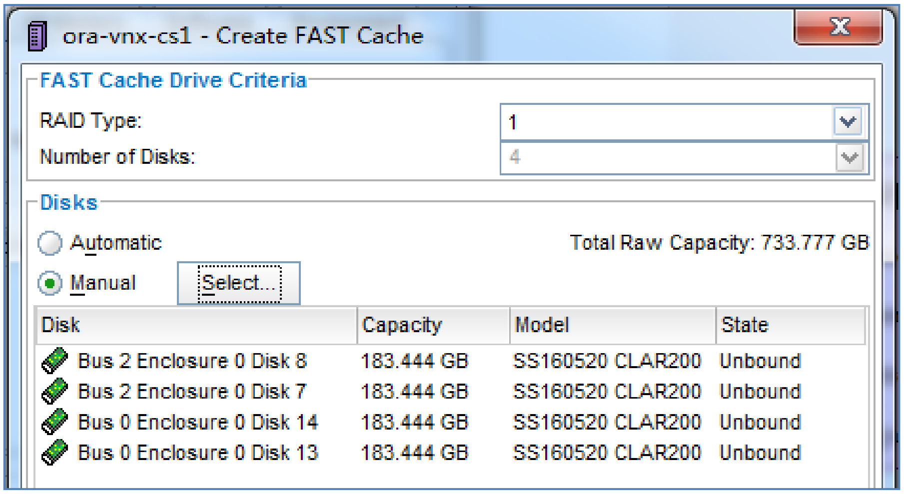


Figure 13. Creating FAST Cache

When the create operation is complete, the storage-system write cache and the FAST Cache are enabled. You can view the current properties of FAST Cache in the **FAST Cache Properties** dialog box, as shown in Figure 14.

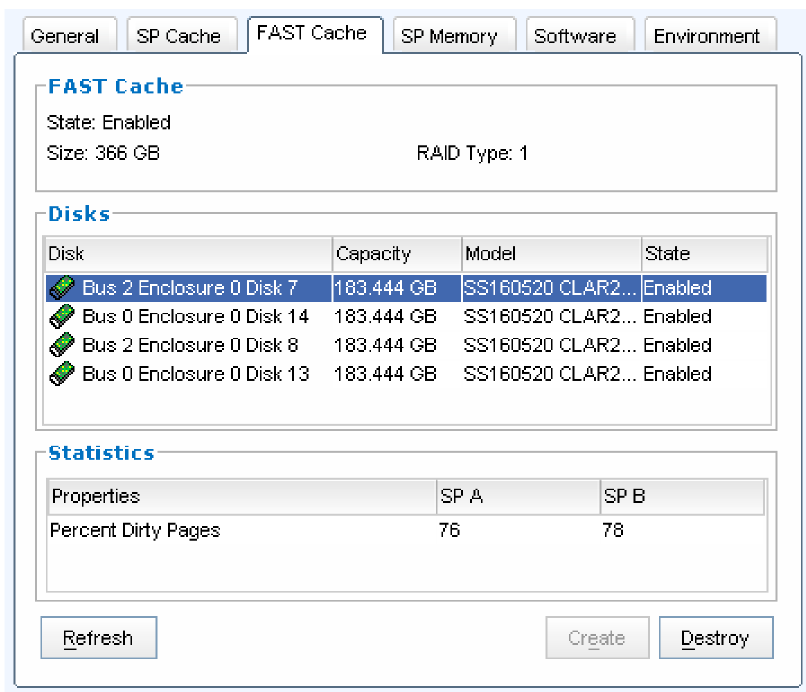


Figure 14. FAST Cache properties

Enable or disable FAST Cache

To enable or disable the FAST Cache for a LUN, display the LUN properties in Unisphere and select or deselect the FAST Cache option, as shown in Figure 15.

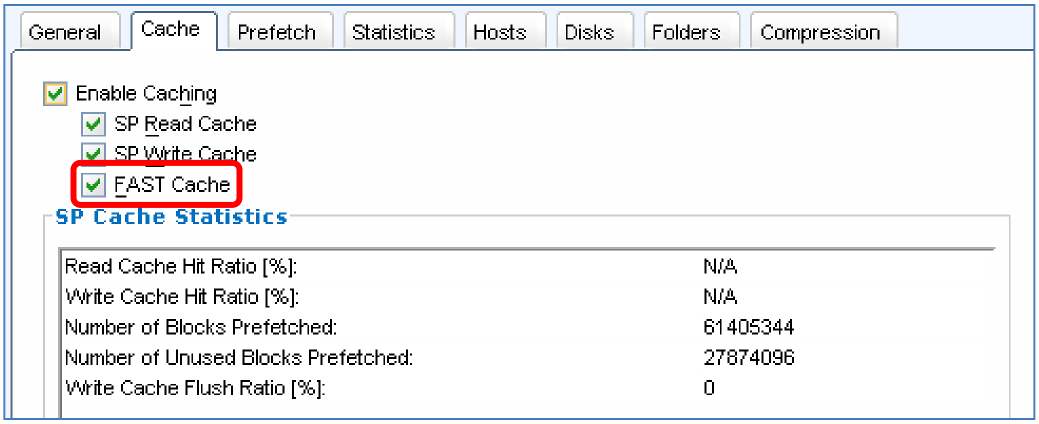


Figure 15. Enable or disable FAST Cache at LUN level

FAST Cache best practices for Oracle

The following best practices should be observed to realize the full advantage of FAST Cache:

- Disable FAST Cache on LUNs that do not require it.
- Size FAST Cache appropriately, depending on the application's active dataset.
- Disable FAST Cache on LUNs where Oracle online redo logs reside.

  Enabling FAST Cache on database online redo logs may or may not help, depending on workload characteristics. The relative gains of enabling FAST Cache on online redo logs will be small when compared to enabling FAST Cache on LUNs with Oracle data files.
- Never enable FAST Cache on archive logs because these files are never overwritten and are rarely read back (unless the database needs to be recovered).

Live migration from physical to virtualized Oracle RAC

Overview

This section describes the migration of the Oracle RAC database from the physical environment to the virtual environment. On the source system, the database resided on an Oracle RAC cluster with two physical nodes. The target platform was two VMware ESX servers, of the same hardware specification as the source nodes. Two virtual machines were created on each server to form a four-node Oracle RAC cluster.

The migration of the database from physical to virtual was nondisruptive, with the database available throughout. The migration process used follows the standard guidelines for adding and deleting cluster nodes and Oracle RAC instances, as described in the following Oracle documentation:

- *Oracle Clusterware Administration and Deployment Guide 11g Release 2 (11.2) – Chapter 4: Adding and Deleting Cluster Nodes*
- *Oracle Real Application Clusters Administration and Deployment Guide 11g Release 2 (11.2) – Chapter 9: Adding and Deleting Oracle RAC from Nodes on Linux and UNIX Systems*

The migration process also used the standard scripts and utilities supplied by Oracle.

Live migration process

The migration from physical to virtual involved the following high-level tasks:

- Start up the virtual machines and configure SSH equivalency on all nodes.
- Configure the private network interconnect.
- Add the virtual machines as cluster nodes.*
- Add Oracle RAC to each virtual machine and add Oracle RAC database instances to each new node.*
- Delete the Oracle RAC physical nodes.

* These processes can be run against multiple nodes at the same time.

Figure 16 shows the point in the physical-to-virtual migration before the physical nodes were deleted—that is, a six-node Oracle RAC made up of two physical nodes and four virtual nodes.

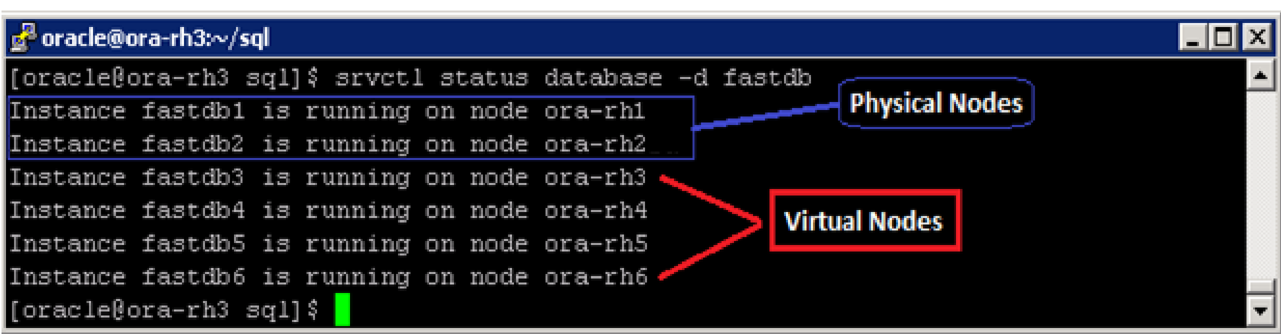


Figure 16. Oracle RAC with physical and virtual nodes

Configure SSH user equivalence

Prior to adding a virtual node, it is essential that Secure Shell (SSH) user equivalence is set up on each node for the oracle user. Establishing user equivalence with SSH provides a secure means of copying files and executing programs on other nodes in the cluster without requiring password prompts.

The process can be carried out manually, or the Oracle script **sshUserSetup.sh** can be used to simplify the process, as outlined here:

1. Before configuring SSH, the **/etc/hosts** file was checked to ensure that all virtual and physical nodes had been added. DNS configuration and resolution was also validated.
2. The **sshUserSetup.sh** script was run on the existing cluster hosts, for user oracle:

   ```
   /home/oracle/sshUserSetup.sh -user <OS User> -hosts "<all
   new hosts>”
   ```

   The following example shows the actual command run on the two physical hosts:

   ```
   /home/oracle/sshUserSetup.sh -user oracle -hosts "ora-rh3
   ora-rh4 ora-rh5 ora-rh6" –noPromptPassphrase
   ```

3. The **sshUserSetup.sh** script was run on all new hosts for user oracle:

   ```
   /home/oracle/sshUserSetup.sh -user <OS User> -hosts "<all
   hosts except self>"
   ```

   For example, on host ora-rh3, the following script was run:

   ```
   /home/oracle/sshUserSetup.sh -user oracle -hosts "ora-rh1 ora-
   rh2 ora-rh4 ora-rh5 ora-rh6" -noPromptPassphrase
   ```

4. When the process was completed, a simple command was run on the remote host to check that users grid and oracle could successfully connect over SSH to all other hosts without a password—for example:

   ```
   /usr/bin/ssh ora-rh3 date
   ```

Configure the private network interconnect

With Oracle RAC, each node requires at least two network adapters—one for the public network interface and one for the private network interface (interconnect).

Highly Available IP (HAIP), a new feature of Oracle Grid Infrastructure 11.2.0.2, was not used for the interconnect. Instead, two network adapters were configured on each physical RAC node and, in the virtual environment, fault tolerance was achieved using a dedicated vNetwork Distributed Switch. See VMware ESX server configuration › Step 1: Create the virtual switches.

Adding the virtual machines as cluster nodes

The virtual machines were added as cluster nodes, following the checks and steps in the manual *Oracle Clusterware Administration and Deployment Guide 11g Release 2 (11.2)*.

For example, to add the first virtual node, the following script was run on physical node ora-rh1 as user oracle:

```
[oracle@ora-rh1 bin]$ cd $ORACLE_HOME/oui/bin
[oracle@ora-rh1 bin]$ ./addNode.sh -silent
"CLUSTER_NEW_NODES={ora-rh3}"
"CLUSTER_NEW_PRIVATE_NODE_NAMES={ora-rh3-priv}"
"CLUSTER_NEW_VIRTUAL_HOSTNAMES={ora-rh3-vip}"
```

When the script successfully completed, and the script output was validated, the **orainstRoot.sh** and **root.sh** scripts were run on the new node as user root.

Adding Oracle RAC database instances

Oracle RAC was added to each virtual machine and Oracle RAC database instances were added to those target nodes, following the steps in the manual *Oracle Real Application Clusters Administration and Deployment Guide 11g Release 2 (11.2)*.

1. The following script was run as user oracle:

   ```
   cd $ORACLE_HOME/oui/bin
   ./addNode.sh -silent "CLUSTER_NEW_NODES={ora-rh3}"
   ```

2. When the script successfully completed, and the script output was validated, the **root.sh** script was run on the new node as user root.
3. From an existing node, as user oracle, the Database Configuration Assistant (DBCA) was run to add an Oracle RAC database instance to each new node:

   ```
   $ORACLE_HOME/bin/dbca -silent -addInstance -nodeList ora-rh3 \
   -gdbName fastdb -instanceName fastdb3 -sysDBAUserName sys \
        -sysDBAPassword <sysdbapwd>
   ```

Deleting the Oracle RAC physical nodes

Oracle database instances were removed from the physical nodes, in line with the manual *Oracle Real Application Clusters Administration and Deployment Guide 11g Release 2 (11.2).*

The following two procedures demonstrate the steps used to remove physical node ora-rh1. These procedures were repeated for ora-rh2.

1. As user oracle, the Oracle SCAN addresses and database services were modified and relocated to use the virtual nodes.
2. The instance was removed by running the **dbca** command from a virtual node as user oracle:

   ```
   dbca -silent -deleteInstance -nodeList ora-rh1 \
      -gdbName fastdb -instanceName fastdb1 -sysDBAUserName sys \
          -sysDBAPassword <sysdbapwd>
   ```

3. As user oracle, the listener was disabled and stopped:

   ```
   srvctl disable listener -n ora-rh1
   srvctl stop listener -n ora-rh1
   ```

4. On the physical node to be removed, the inventory was updated:

```
./runInstaller -updateNodeList ORACLE_HOME=$ORACLE_HOME \
     "CLUSTER_NODES={ ora-rh1 }" -local
```

5. The Oracle home was then uninstalled from that node:

```
$ORACLE_HOME/deinstall/deinstall -local
```

6. The following command was run from a virtual node to update the inventory:

```
./runInstaller -updateNodeList ORACLE_HOME=$ORACLE_HOME \
    "CLUSTER_NODES={ ora-rh3, ora-rh4, ora-rh5, ora-rh6 }"
```

When the Oracle RAC database binaries and instance had been removed (see steps 1 to 6 above), the physical servers were removed as cluster nodes, following the checks and steps in the manual *Oracle Clusterware Administration and Deployment Guide 11g Release 2 (11.2).*

1. Clusterware applications and daemons were disabled on the physical node to be removed:

```
$ORACLE_HOME/crs/install/rootcrs.pl -deconfig -force
```

2. From one of remaining virtual nodes (as user root), the physical node was deleted from the Clusterware:

```
crsctl delete node -n ora-rh1
```

3. As user oracle, from the physical node to be removed, the inventory was updated:

```
/u01/app/11.2.0.2/grid/oui/bin/runInstaller -updateNodeList
 ORACLE_HOME=/u01/app/11.2.0.2/grid "CLUSTER_NODES={ora-rh1}"
   CRS=TRUE -local
```

4. Clusterware software was uninstalled from the physical node:

```
/u01/app/11.2.0.2/grid/deinstall/deinstall -local
```

5. From a virtual node, the Clusterware was updated with the existing nodes:

```
./runInstaller -updateNodeList ORACLE_HOME=$ORACLE_HOME \
   "CLUSTER_NODES={ora-rh3, ora-rh4, ora-rh5, ora-rh6}" CRS=TRUE
```

6. It was then verified that the node has been removed and the remaining nodes were valid:

```
cluvfy stage -post nodedel -n ora-rh1 -verbose
```

Testing and validation

FAST Cache and manual tiering comparison

In 2008, EMC introduced Enterprise Flash Drive technology to its midrange storage arrays as a high-performing storage tier. VNX storage provides the option to use Flash drives as a separate tier within a manual tiering strategy, as part of a pool of storage managed automatically by FAST VP, and/or as FAST Cache.

Manual tiering involves a repeated process that takes 9 hours or more to complete each time. In contrast, both FAST VP and FAST Cache operate automatically, eliminating the need for manually identifying and moving or caching hot data. As shown in Figure 17, configuring FAST Cache is a one-off process taking 50 minutes or less, and hot and cold data is then cached in and out of FAST Cache continuously and automatically.

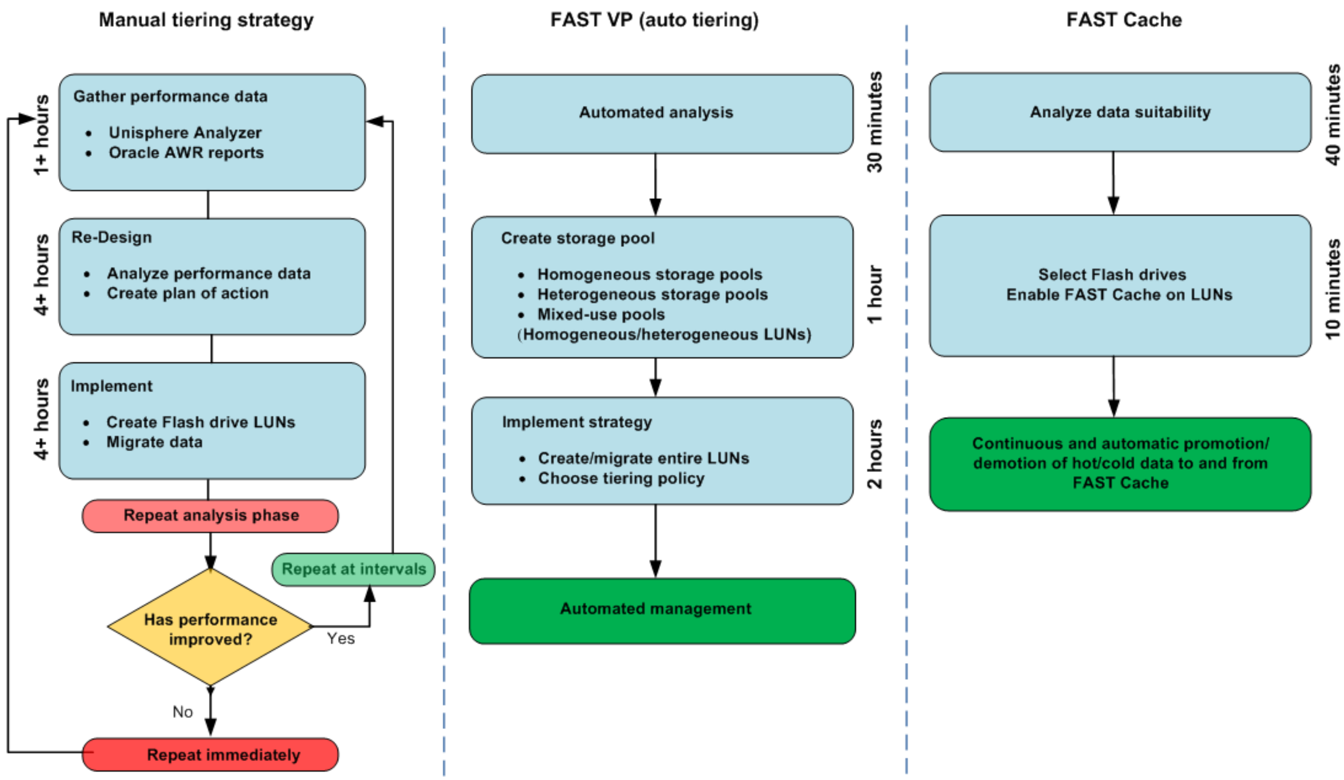


Figure 17. FAST Cache and manual tiering comparison

FAST Cache test procedure

To test the effects of FAST Cache, a Swingbench Order Entry workload was generated and run against schema SOE with and without FAST Cache enabled. The same test procedure was carried out in both the physical and virtual environments.

- The baseline was established on an all SAS drive configuration. The database was 1 TB in size, with the data files deployed on two file systems spread over eight LUNs, and the online redo logs deployed on two LUNs.
- A workload was generated against the schema and was scaled from 600 to 2,000 users.
- The performance of the database was monitored, and the average front-end IOPS and database TPM were recorded for each user iteration.

- Four 200 GB Flash drives on the VNX were then configured for FAST Cache in a RAID 1 configuration. This provided a FAST Cache of 400 GB. FAST Cache was enabled for the eight LUNs that contained the database data files.
- Workload was again generated against the schema and was scaled from 600 to 2,000 users.
- The performance metrics from the FAST Cache environment were compared with those from the baseline SAS-only environment to determine how the database performed in each scenario and the improvement that could be achieved by using FAST Cache.

Notes

- Benchmark results are highly dependent upon workload, specific application requirements, and system design and implementation. Relative system performance will vary as a result of these and other factors. Therefore, this workload should not be used as a substitute for a specific customer application benchmark when critical capacity planning and/or product evaluation decisions are contemplated.
- All performance data contained in this report was obtained in a rigorously controlled environment. Results obtained in other operating environments may vary significantly.
- EMC Corporation does not warrant or represent that a user can or will achieve similar performance expressed in transactions per minute.

FAST Cache read/write hit ratio

Figure 18 tracks the FAST Cache I/O hit ratio of the database LUNs.

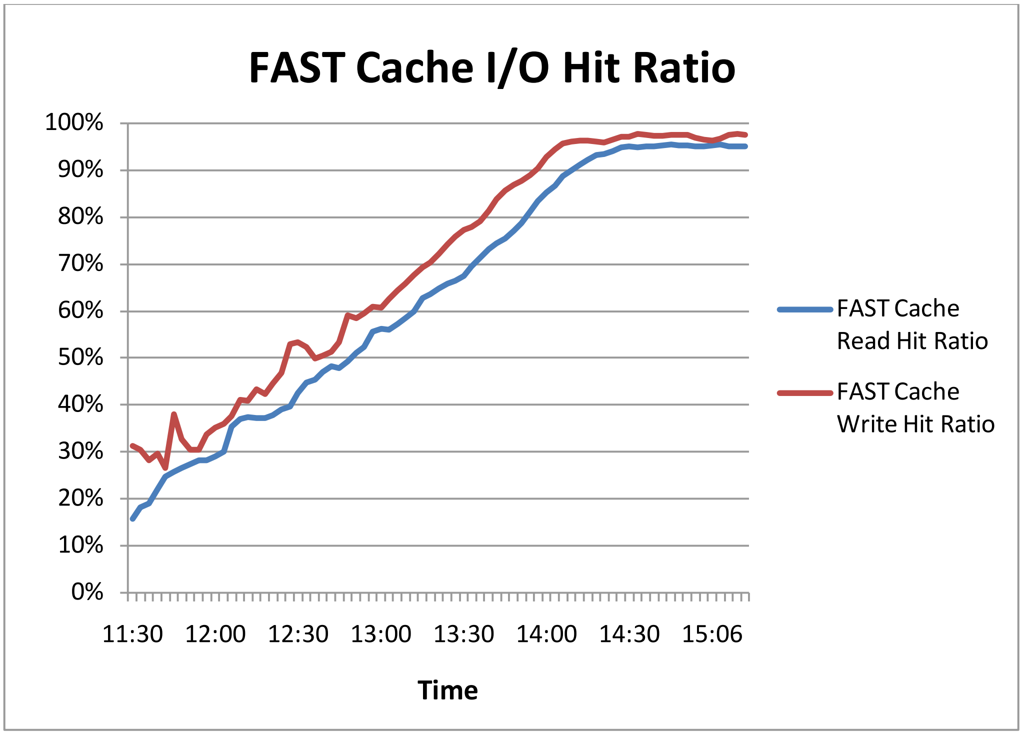


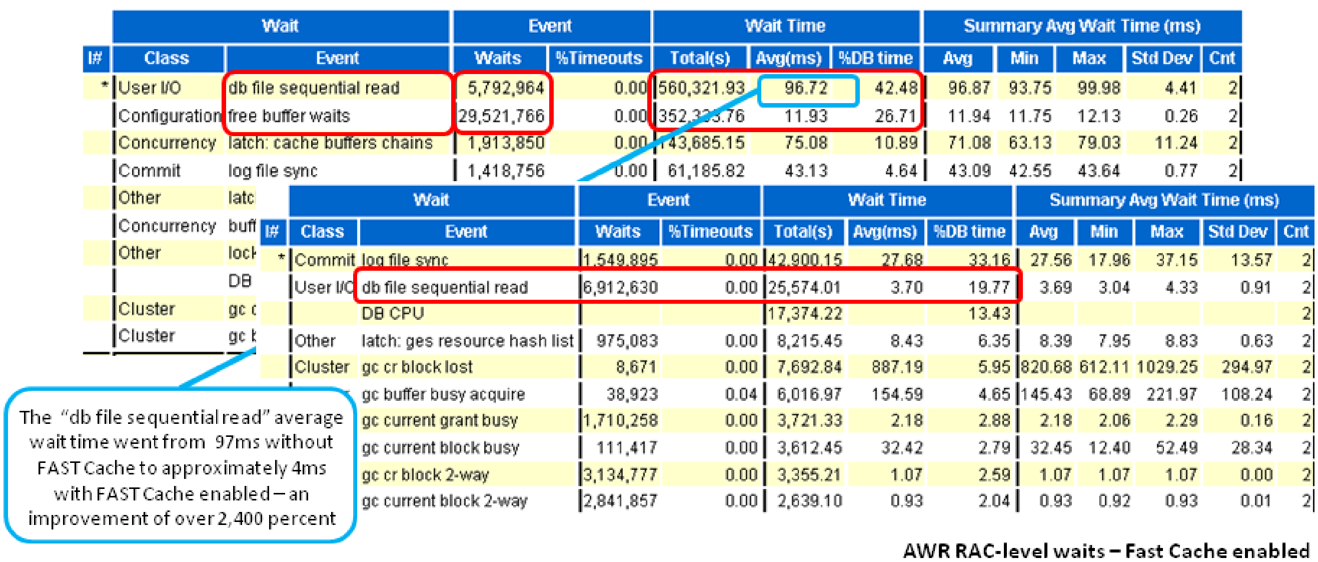
Figure 18. FAST Cache I/O hit ratio

FAST Cache takes some time to warm up before it achieves optimal performance improvement. When FAST Cache is initially created it is empty. During the warm-up period, as more and more hot data is cached, the FAST Cache hit rate increases gradually. In this solution, after a warm-up period of 2.5 hours, the read and write hit ratios were almost 100 percent.

Wait statistics from Oracle AWR reports

Oracle foreground wait statistics highlight potential bottlenecks in Oracle RAC environments. Figure 19 and Figure 20 show extracts from the Automatic Workload Repository (AWR) RAC reports and compare the waits for the baseline and FAST Cache-enabled tests in both the physical and virtual RAC environments respectively.

In Figure 19, the physical environment AWR report identifies **db file sequential read** and **free buffer waits** as the top two wait events during the baseline testing.



AWR RAC-level waits – Baseline (without FAST Cache)

I#	Class	Event	Waits	%Timeouts	Total(s)	Avg(ms)	%DB time	Avg	Min	Max	Std Dev	Cnt
	Wait		Event		Wait Time			Summary Avg Wait Time (ms)				
*	User I/O	db file sequential read	5,792,964	0.00	560,321.93	96.72	42.48	96.87	93.75	99.98	4.41	2
	Configuration	free buffer waits	29,521,766	0.00	352,3[illegible]3.76	11.93	26.71	11.94	11.75	12.13	0.26	2
	Concurrency	latch: cache buffers chains	1,913,850	0.00	143,685.15	75.08	10.89	71.08	63.13	79.03	11.24	2
	Commit	log file sync	1,418,756	0.00	61,185.82	43.13	4.64	43.09	42.55	43.64	0.77	2
	Other	latc										
	Concurrency	buff										
	Other	lock										
		DB										
	Cluster	gc c										
	Cluster	gc b										

AWR RAC-level waits – Fast Cache enabled

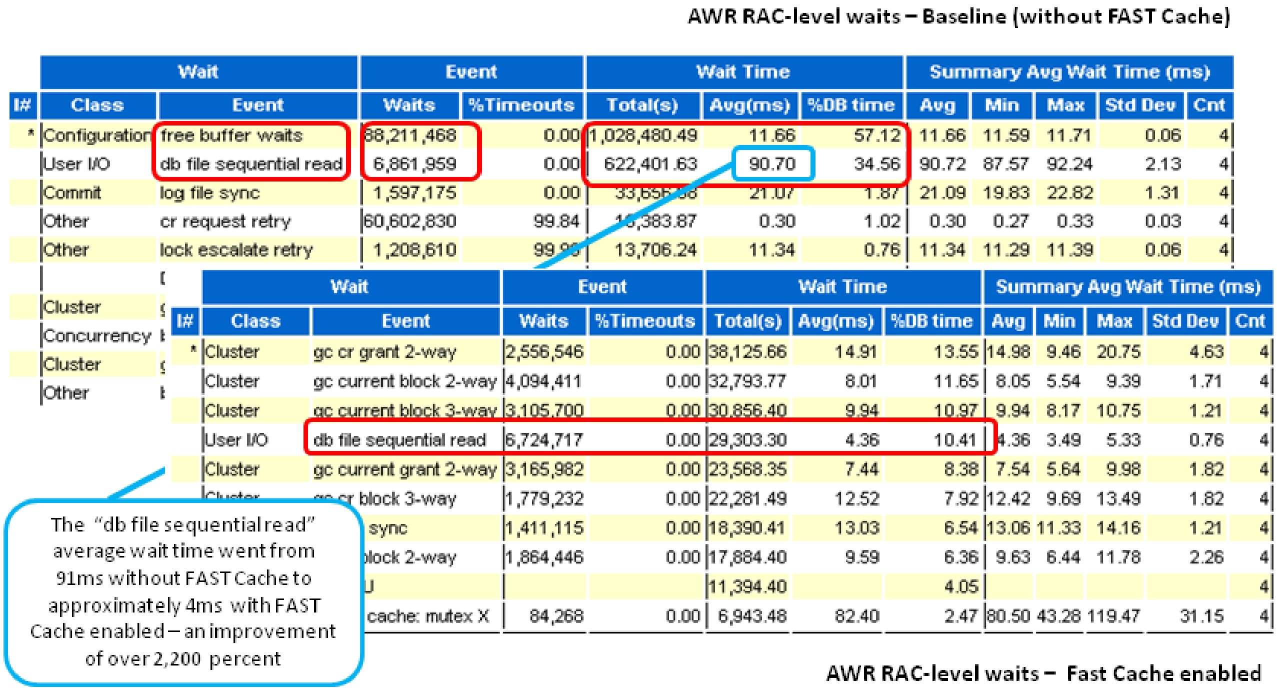
I#	Class	Event	Waits	%Timeouts	Total(s)	Avg(ms)	%DB time	Avg	Min	Max	Std Dev	Cnt
	Wait		Event		Wait Time			Summary Avg Wait Time (ms)				
*	Commit	log file sync	1,549,895	0.00	42,900.15	27.68	33.16	27.56	17.96	37.15	13.57	2
	User I/O	db file sequential read	6,912,630	0.00	25,574.01	3.70	19.77	3.69	3.04	4.33	0.91	2
		DB CPU			17,374.22		13.43					2
	Other	latch: ges resource hash list	975,083	0.00	8,215.45	8.43	6.35	8.39	7.95	8.83	0.63	2
	Cluster	gc cr block lost	8,671	0.00	7,692.84	887.19	5.95	820.68	612.11	1029.25	294.97	2
		gc buffer busy acquire	38,923	0.04	6,016.97	154.59	4.65	145.43	68.89	221.97	108.24	2
		gc current grant busy	1,710,258	0.00	3,721.33	2.18	2.88	2.18	2.06	2.29	0.16	2
		gc current block busy	111,417	0.00	3,612.45	32.42	2.79	32.45	12.40	52.49	28.34	2
		gc cr block 2-way	3,134,777	0.00	3,355.21	1.07	2.59	1.07	1.07	1.07	0.00	2
		gc current block 2-way	2,841,857	0.00	2,639.10	0.93	2.04	0.93	0.92	0.93	0.01	2

Figure 19. AWR reports for physical RAC

Db file sequential read events indicate a possible I/O bottleneck. **Free buffer waits** usually indicate that the database writer (DBWR) is not fast enough in clearing dirty blocks from the buffer cache to disk.

When FAST Cache is enabled, **free buffer waits** is no longer reported as a top wait event and the bottleneck from **db file sequential read** events is reduced to 4.6 percent of the baseline result. The total time for the Top 10 waits is only 22 percent of the **db file sequential read** wait time for the baseline, while the transaction rate has improved by more than 100 percent (see FAST Cache effects on transactions per minute).

Figure 20 shows the AWR reports for the virtual environment and a comparison between the baseline and FAST Cache-enabled tests. The events highlighted are again indications of potential bottlenecks within the virtual environment—that is, **free buffer waits** and **db file sequential read**.



AWR RAC-level waits – Baseline (without FAST Cache)

	Wait		Event		Wait Time			Summary Avg Wait Time (ms)				
I#	Class	Event	Waits	%Timeouts	Total(s)	Avg(ms)	%DB time	Avg	Min	Max	Std Dev	Cnt
*	Configuration	free buffer waits	88,211,468	0.00	1,028,480.49	11.66	57.12	11.66	11.59	11.71	0.06	4
	User I/O	db file sequential read	6,861,959	0.00	622,401.63	90.70	34.56	90.72	87.57	92.24	2.13	4
	Commit	log file sync	1,597,175	0.00	[illegible]	21.07	1.87	21.09	19.83	22.82	1.31	4
	Other	cr request retry	60,602,830	99.84	[illegible]	0.30	1.02	0.30	0.27	0.33	0.03	4
	Other	lock escalate retry	1,208,610	[illegible]	13,706.24	11.34	0.76	11.34	11.29	11.39	0.06	4
		[illegible]										
	Cluster	[illegible]										
	Concurrency	[illegible]										
	Cluster	[illegible]										
	Other	[illegible]										

AWR RAC-level waits – Fast Cache enabled

	Wait		Event		Wait Time			Summary Avg Wait Time (ms)				
I#	Class	Event	Waits	%Timeouts	Total(s)	Avg(ms)	%DB time	Avg	Min	Max	Std Dev	Cnt
*	Cluster	gc cr grant 2-way	2,556,546	0.00	38,125.66	14.91	13.55	14.98	9.46	20.75	4.63	4
	Cluster	gc current block 2-way	4,094,411	0.00	32,793.77	8.01	11.65	8.05	5.54	9.39	1.71	4
	Cluster	gc current block 3-way	3,105,700	0.00	30,856.40	9.94	10.97	9.94	8.17	10.75	1.21	4
	User I/O	db file sequential read	6,724,717	0.00	29,303.30	4.36	10.41	4.36	3.49	5.33	0.76	4
	Cluster	gc current grant 2-way	3,165,982	0.00	23,568.35	7.44	8.38	7.54	5.64	9.98	1.82	4
	Cluster	gc cr block 3-way	1,779,232	0.00	22,281.49	12.52	7.92	12.42	9.69	13.49	1.82	4
		sync	1,411,115	0.00	18,390.41	13.03	6.54	13.06	11.33	14.16	1.21	4
		block 2-way	1,864,446	0.00	17,884.40	9.59	6.36	9.63	6.44	11.78	2.26	4
		U			11,394.40		4.05					4
		cache: mutex X	84,268	0.00	6,943.48	82.40	2.47	80.50	43.28	119.47	31.15	4

Figure 20. AWR reports for virtual RAC

When Fast Cache is enabled, the bottleneck from **db file sequential read** events is reduced to only 4.7 percent of the baseline result. The total time for the Top 10 waits is 37 percent of the **db file sequential read** wait time for the baseline, while the transaction rate has improved by more than 100 percent (see FAST Cache effects on transactions per minute). Also, **free buffer waits** is no longer a top wait event.

FAST Cache effects on transactions per minute

Figure 21 and Figure 22 compare the TPM recorded during testing, with and without FAST Cache enabled.

The charts show that, for all user workloads, in both the physical and virtual environments, the number of transactions processed was significantly higher when FAST Cache was enabled. Under the maximum load tested, FAST Cache boosted performance by more than 100 percent.

In the physical environment, the performance metrics for 1,800 users were:

- TPM without FAST Cache - 185,941
- TPM with FAST Cache - 402,226

This shows a 116 percent improvement in performance (see Figure 21).

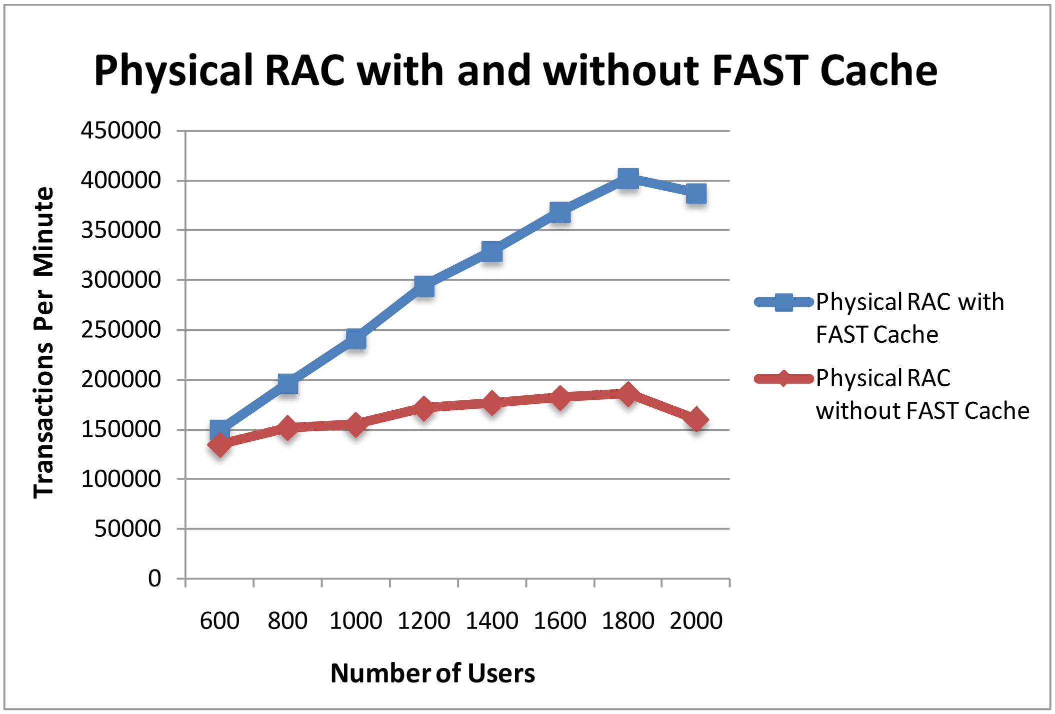


Figure 21. Physical RAC with and without FAST Cache

In the virtual environment, the performance metrics for 1,800 users were:

- TPM without FAST Cache - 172,679
- TPM with FAST Cache - 354,611

This shows a 105 percent improvement in performance (see Figure 22).

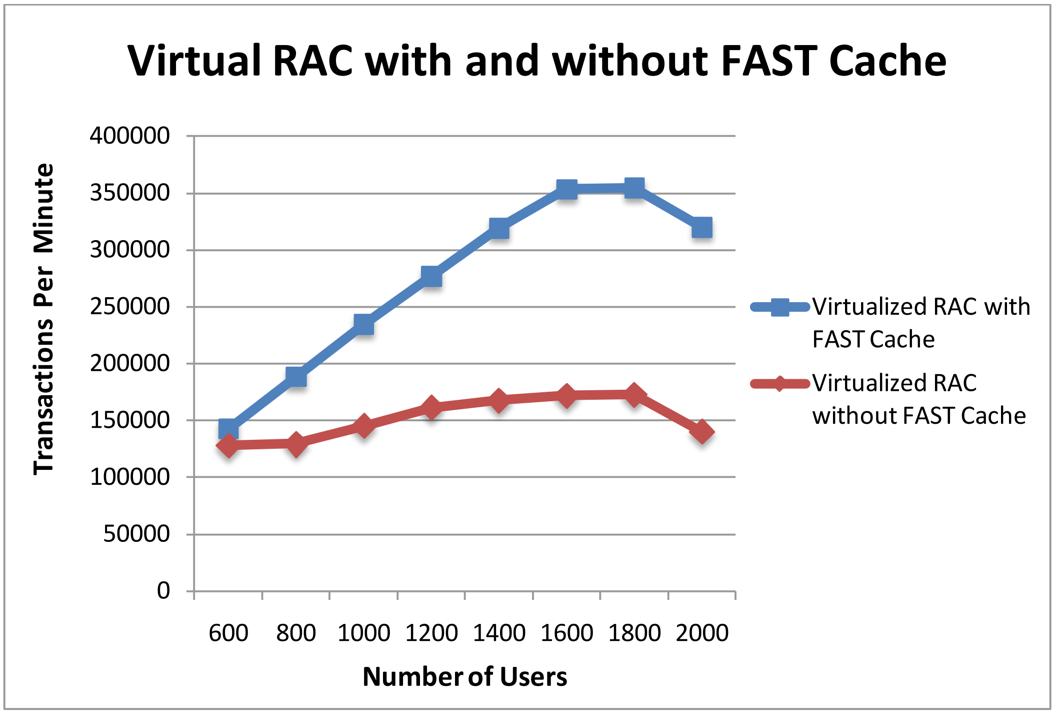


Figure 22. Virtual RAC with and without FAST Cache

The Oracle database was created with minimal changes to the default settings. No changes were made to the database between the baseline and FAST Cache testing. All performance improvements were a direct result of enabling FAST Cache. Further performance improvement with user scaling could be achieved by tuning the Oracle database and the application/schema.

FAST Cache effects on response times

Figure 23 compares the Swingbench response times at 1,800 users, with and without FAST Cache enabled.

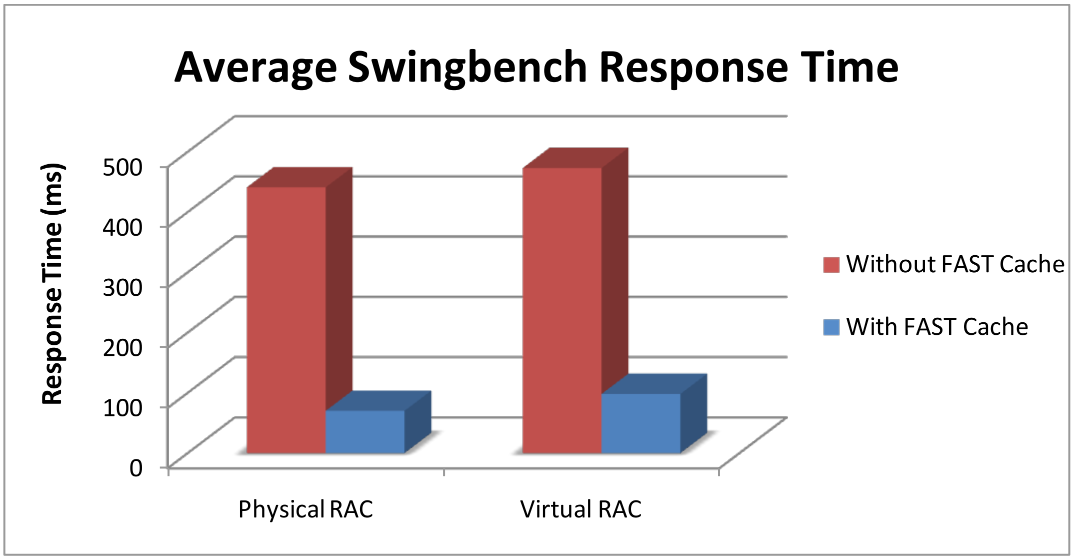


Figure 23. Swingbench response times with and without FAST Cache

When FAST Cache was enabled, and after the hot data was promoted, response times delivered an impressive overall improvement of 84 percent for the physical environment and 79 percent for the virtual environment.

The reduced response times allow the application to scale much better and improve the end-user experience significantly.

FAST Cache effects on IOPS

Figure 24 shows the increase in average IOPS for the data file systems. In both the physical and virtual environments, as more and more hot data was cached in by FAST Cache, a 170 percent improvement in IOPS was observed.

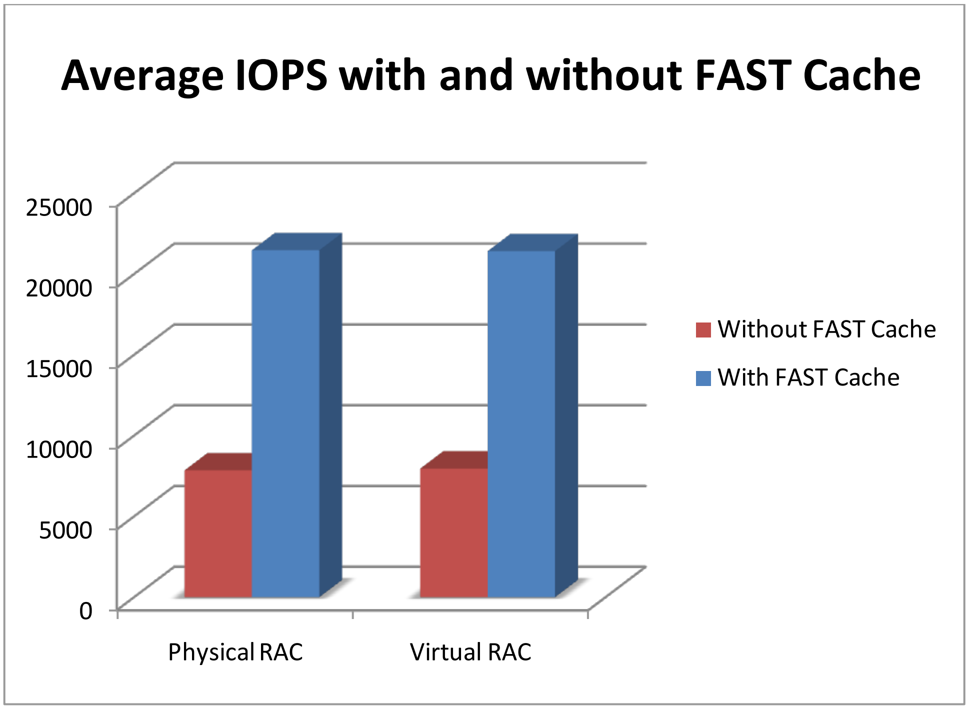


Figure 24. Average IOPS with and without FAST Cache

Conclusion

Summary

The testing detailed in the paper demonstrates that when FAST Cache is introduced into the physical and virtual Oracle RAC OLTP environments, it reduces I/O accesses to the HDDs and directs them to the Flash drives, which dramatically increases the OLTP throughput and maintains very low response times. The overall application performance improves significantly as a result.

FAST Cache technology creates a faster medium, on Flash drives, for accessing frequently accessed data at lower latencies. Hot data is cached in and cold data flushed out of FAST Cache automatically and transparently, depending on data usage patterns. This eliminates the need for administrators to manually classify the hot and cold data.

Another important FAST Cache benefit is improved TCO. Using FAST Cache reduces the I/O to the back-end HDDs. This means that an existing set of HDDs can deliver the performance typically provided by a faster drive configuration, such as more HDDs or a different RAID type. In fact, over a period of time, the number of faster SAS drives may be reduced or replaced with slower NL-SAS drives, while maintaining the same application performance.

Findings

The key findings of the testing performed for the solution demonstrate that:

- By creating a FAST Cache with just four Flash drives, the performance of transactions per minute improved by over 100 percent for both the physical and virtual environments.
- Enabling FAST Cache improved the average response time by 84 percent in the physical environment, and by 79 percent in the virtual environment.
- Using FAST Cache as a secondary cache delivered a 170 percent improvement in IOPS.
- FAST Cache serviced approximately 95 percent of the read and write IOPS in both the physical and virtual environments. FAST Cache misses could still be a cache hit if the data is in the storage processor (SP) cache.
- Live migration of the Oracle RAC 11*g* R2 database from a physical to a virtual environment was achieved without loss of service.

References

White papers

For additional information, see the EMC white papers listed below.

- *Deploying Oracle Database Applications on EMC VNX Unified Storage*
- *EMC CLARiiON, Celerra Unified, and FAST Cache—A Detailed Review*
- *Oracle E-Business Suite Deployment Agility Enabled by EMC Unified Storage and VMware—An Architectural Overview*
- *Leveraging EMC FAST Cache with Oracle OLTP Database Applications—Applied Technology*
- *Optimizing EMC Celerra IP Storage on Oracle 11g Direct NFS—Applied Technology*
- *Maximize Operational Efficiency for Oracle RAC with EMC Symmetrix FAST VP (Automated Tiering) and VMware vSphere—An Architectural Overview*

Product documentation

For additional information, see the EMC product documents listed below.

- *Unisphere Help* in the Unisphere GUI
- *Configuring Standbys on VNX*

Other documentation

For additional information, see the documents listed below.

- *Oracle Real Application Clusters Installation Guide 11g Release 2 (11.2) for Linux*
- *Oracle Real Application Clusters Administration and Deployment Guide 11g Release 2 (11.2)*
- *Oracle Grid Infrastructure Installation Guide 11g Release 2 (11.2) for Linux*
- *Oracle Clusterware Administration and Deployment Guide 11g Release 2 (11.2)*
- *My Oracle Support Note 1212703.1*
- *Support Position for Oracle Products Running on VMware Virtualized Environments [ID 249212.1]* published on 08 November 2010
- *Virtualizing Oracle E-Business Suite on Vblock Infrastructure Platforms* (www.vce.com/pdf/solutions/vce-oracle-ebs-p2v-migration.pdf)