TECHNICAL REPORT

Nimble Storage Best Practices Guide: Oracle Database on Oracle Linux & RHEL 6

Document Revision

Table 1.

Date	Revision	Description
1/9/2012	1.0	Initial Draft
6/13/2013	1.1	Revised
7/15/2013	1.2	Revised

THIS TECHNICAL TIP 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.

Nimble Storage: All rights reserved. Reproduction of this material in any manner whatsoever without the express written permission of Nimble is strictly prohibited.

Table of Contents

Introduction

The purpose of this technical white paper is to describe the best practices for implementing Oracle databases on Nimble Storage running on Oracle Linux operating system.

Oracle performance tuning is beyond the scope of this paper. Please visit www.oracle.com for *Oracle Performance Tuning Guide* for more information in tuning your database.

Audience

This guide is intended for Oracle database solution architects, storage engineers, system administrators and IT managers who analyze, design and maintain a robust database environment on Nimble Storage. It is assumed that the reader has a working knowledge of iSCSI SAN network design, and basic Nimble Storage operations. Knowledge of Oracle Linux operating system, Oracle Clusterware, and Oracle database is also required.

Scope

During the design phase for a new Oracle database implementation, DBAs and Storage Administrators often times work together to come up with the best storage needs. They have to consider many storage configuration options to facilitate high performance and high availability. In order to protect data against failures of disk drives, host bus adapters (HBAs), and switches, they need to consider using different RAID levels and multiple paths. When you have different RAID levels come into play for performance, TCO tends to increase as well. For example, in order to sustain a certain number of IOPS with low latency for an OLTP workload, DBAs would require a certain number of 15K disk drives with RAID 10. The higher the number of required IOPS, the more 15K drives are needed. The reason is because mechanical disk drives have seek times and transfer rate, therefore, you would need more of them to handle the required IOPS with acceptable latency. This will increase the TCO tremendously over time. Not to mention that if the database is small in capacity but the required IOPS is high, you would end up with a lot of wasted space in your SAN.

This white paper explains the Nimble technology and how it can lower the TCO of your Oracle environment and still achieve the performance required. This paper also discusses the best practices for implementing Oracle databases on Nimble Storage.

Nimble Storage Features

Cache Accelerated Sequential Layout (CASL™)

Nimble Storage arrays are the industry's first flash-optimized storage designed from the ground up to maximize efficiency. CASL accelerates applications by using flash as a read cache coupled with a write-optimized data layout. It offers high performance and capacity savings, integrated data protection, and easy lifecycle management.

Flash-Based Dynamic Cache

Accelerate access to application data by caching a copy of active "hot" data and metatdata in flash for reads. Customers benefit from high read throughput and low latency.

Write-Optimized Data Layout

Data written by a host is first aggregated or coalesced, then written sequentially as a full stripe with checksum and RAID parity information to a pool of disk; CASL's sweeping process also consolidates freed up disk space for future writes. Customers benefit from fast sub-millisecond writes and very efficient disk utilization

Inline Universal Compression

Compress all data inline before storing using an efficient variable-block compression algorithm. Store 30 to 75 percent more data with no added latency. Customers gain much more usable disk capacity with zero performance impact.

Instantaneous Point-in-Time Snapshots

Take point-in-time copies, which do not require data to be copied on future changes (redirect-on-write). Fast restores without copying data. Customers benefit from a single, simple storage solution for primary and secondary data, frequent and instant backups, fast restores and significant capacity savings.

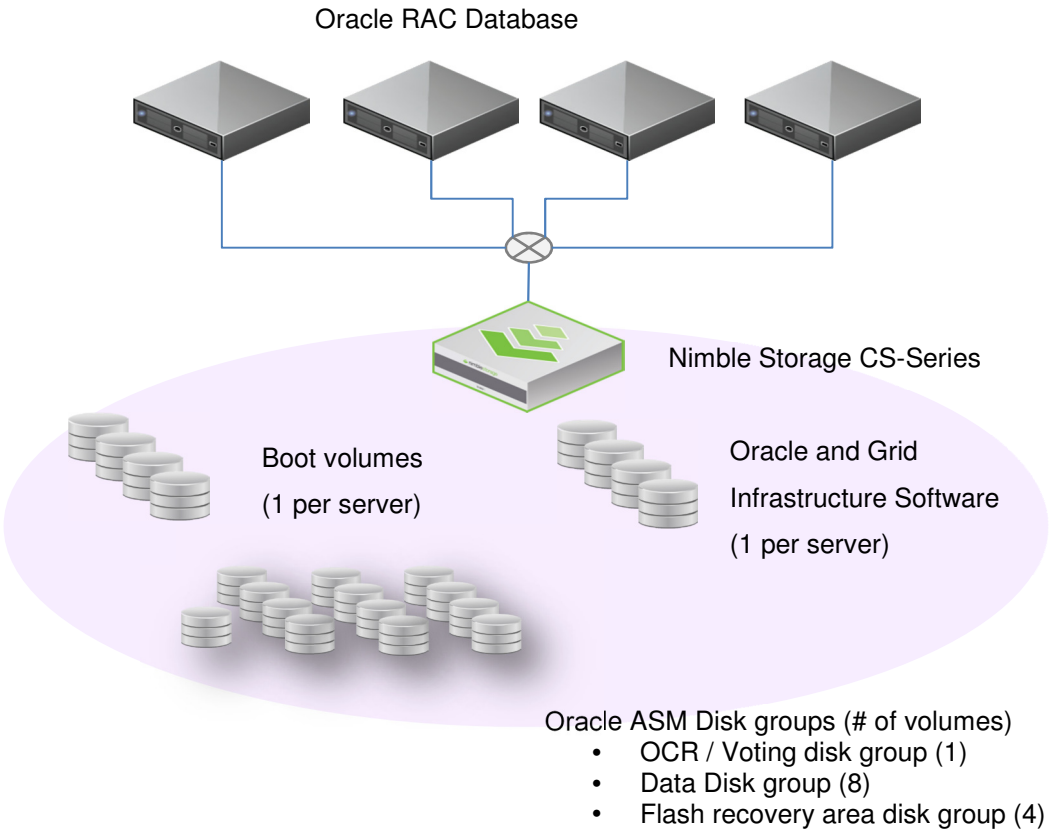
Efficient Integrated Replication

Maintain a copy of data on a secondary system by only replicating compressed changed data on a set schedule. Reduce bandwidth costs for WAN replication and deploy a disaster recovery solution that is affordable and easy to manage.

Zero-Copy Clones

Instantly create full functioning copies or clones of volumes. Customers get great space efficient and performance on cloned volumes, making them ideal for test, development, and staging Oracle databases.

Oracle Database on Oracle Linux with Nimble Storage



When considering best practices for running Oracle databases including RAC on Oracle Linux, the areas to consider include performance, data protection and efficiency –especially as it related to test and development. This document covers the best practices including performance setting and volume setup with Oracle ASM.

Performance Settings

When running Oracle database on Linux, there are many operating system settings that need to be tweaked to get the best performance and uptime. However, not all settings will make the Oracle database perform better. For an optimal performing database, there are many factors that need to be looked at. Such factors include, but not limited to:

- How the application was written to access the database data?
- Are the queries optimal?
- Are the logical database structures layout optimal for the workload (i.e. indexes, table partitioning)?
- What is the Server CPUs and memory profile?
- What type of IO Scheduler being used in Linux?
- What is the Queue depth setting?
- What File system is being used?
- What is the IO size chosen?
- How many Volumes/LUNs are created on storage?
- What is the number of IO paths to storage?

Nimble Recommended Settings

Not all settings are optimal for all workloads. Certain settings work well for transactional (OLTP) workload but not for analytical (DSS) workload and vice versa. This paper discusses the optimal settings for both OLTP and DSS workloads.

OLTP Workload Settings

For database systems support OLTP workloads, these settings need to be in place for optimal performance.

- Nimble Array
  - Nimble OS should be at least 1.4.5.0
- Linux Operating System
  - iSCSI
    - node.session.timeo.replacement_timeout = 10
    - node.conn[0].timeo.noop_out_interval = 5
    - node.conn[0].timeo.noop_out_timeout = 10
  - Multipath
    - Use aliases for mapped LUNs
    - path_grouping_policy = group_by_serial
    - features = "1 queue_if_no_path"
    - path_selector = round-robin 0"
    - path_checker = tur
    - rr_min_io_rq = 1000
    - rr_weight = priorities
    - failback = immediate
  - IO Scheduler
    - noop
  - CPU Governor
    - performance
  - Data Network
    - 2 separate subnets
    - 2 x iSCSI NICs
    - 10GigE for iSCSI networks
    - Use jumbo frames (MTU 9000) for iSCSI networks
- Oracle database on ASM
  - 2 ASM disk groups (1 for database data and 1 for online redo logs)
  - 8 volumes minimum for database data disk group using 8KB Nimble performance policy
  - 4 volumes minimum for online redo logs disk group using 4KB Nimble performance policy
  - Use Oracle 8KB block size for database creation
  - Use Oracle ASM with External Redundancy when creating diskgroups
    - Consider leveraging Oracle ASM normal redundancy to safeguard against OS related failures in accessing ASM volumes. This would result in mirror copies and twice the space usage versus external redundancy.
  - Set filesystemio_options=setall (Direct I/O and Async I/O)
  - Create online redo logs with 4KB block size. Set the hidden parameter "_disk_sector_size_override=TRUE" for the database.
  - Increase the number of db_writer_processes

- Oracle database on EXT file system
    - Use LVM
    - 2 volume groups (1 VG for database data and 1 VG for online redo logs)
    - 8 volumes minimum for database data VG using 8KB Nimble performance policy
    - 4 volumes minimum for online redo logs VG using 4KB Nimble performance policy
    - Use Oracle 8KB block size for database creation
    - Set "filesystemio_options=setall" (Allows Direct I/O and Async I/O)
    - Create online redo logs with 4KB block size. Set the hidden parameter "_disk_sector_size_override=TRUE" for the database.
    - Increase the number of db_writer_processes

  Please refer to the Nimble Linux 6 best practices guide for more information on using LVM and creating EXT file system and mount options.

DSS Workload Settings

For database systems support DSS workloads, these settings need to be in place for optimal performance.

- Nimble Array
    - Nimble OS should be at least 1.4.5.0
- Linux Operating System
    - iSCSI
        - node.session.timeo.replacement_timeout = 10
        - node.conn[0].timeo.noop_out_interval = 5
        - node.conn[0].timeo.noop_out_timeout = 10
    - Multipath
        - Use aliases for mapped LUNs
        - path_grouping_policy = group_by_serial
        - features = "1 queue_if_no_path"
        - path_selector = round-robin 0"
        - path_checker = tur
        - rr_min_io_rq = 1000
        - rr_weight = priorities
        - failback = immediate
    - IO Scheduler
        - Noop
    - CPU Governor
        - performance
    - Data Network
        - 2 separate subnets
        - 2 x iSCSI NICs
        - 10GigE for iSCSI networks
        - Use jumbo frames (MTU 9000) for iSCSI networks

- Oracle database on ASM
    - 2 ASM disk groups (1 for database data and 1 for online redo logs)
    - 8 volumes minimum for database data disk group using 32KB Nimble performance policy
    - 4 volumes minimum for online redo logs disk group using 4KB Nimble performance policy

- Use Oracle 32KB block size for database creation
- Use Oracle ASM with External Redundancy when creating diskgroups
  - Consider leveraging Oracle ASM normal redundancy to safeguard against OS related failures in accessing ASM volumes. This would result in mirror copies and twice the space usage versus external redundancy.
- Set filesystemio_options=setall (Direct I/O and Async I/O)
- Create online redo logs with 4KB block size. Set the hidden parameter "_disk_sector_size_override=TRUE" for the database.
  Increase the number of db_writer_processes

- Oracle database on EXT file system
  - Use LVM
  - 2 volume groups (1 VG for database data and 1 VG for online redo logs)
  - 8 volumes minimum for database data VG using 32KB Nimble performance policy
  - 4 volumes minimum for online redo logs VG using 4KB Nimble performance policy
  - Use Oracle 32KB block size for database creation
  - Set "filesystemio_options=setall" (Allows Direct I/O and Async I/O)
  - Create online redo logs with 4KB block size. Set the hidden parameter "_disk_sector_size_override=TRUE" for the database.
  - Increase the number of db_writer_processes

  Please refer to the Nimble Linux 6 best practices guide for more information on using LVM and creating EXT file system and mount options.

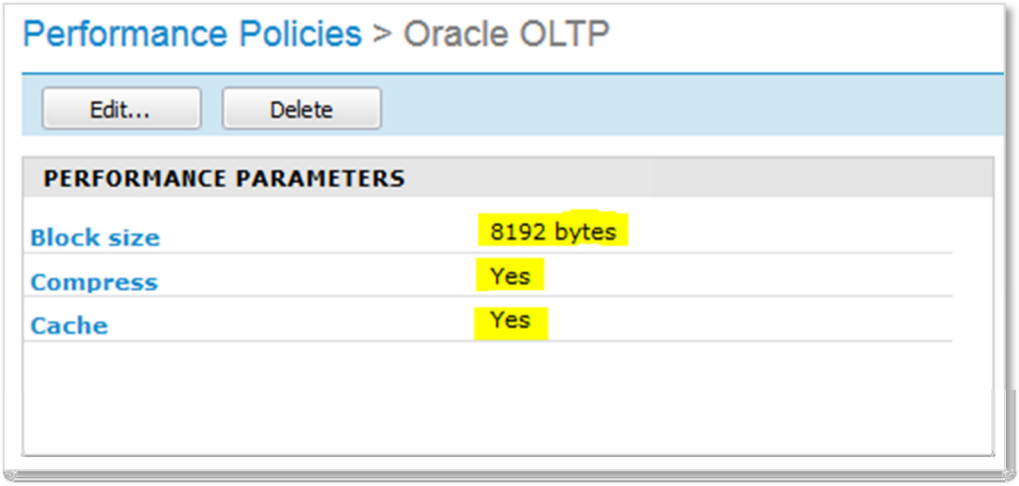
Creating Nimble Volumes for Oracle DB with ASM

Oracle Automatic Storage Management introduced in Oracle 10g simplifies the storage of Oracle data files, control and log files. The tables below show the recommended settings for OLTP and DSS database.

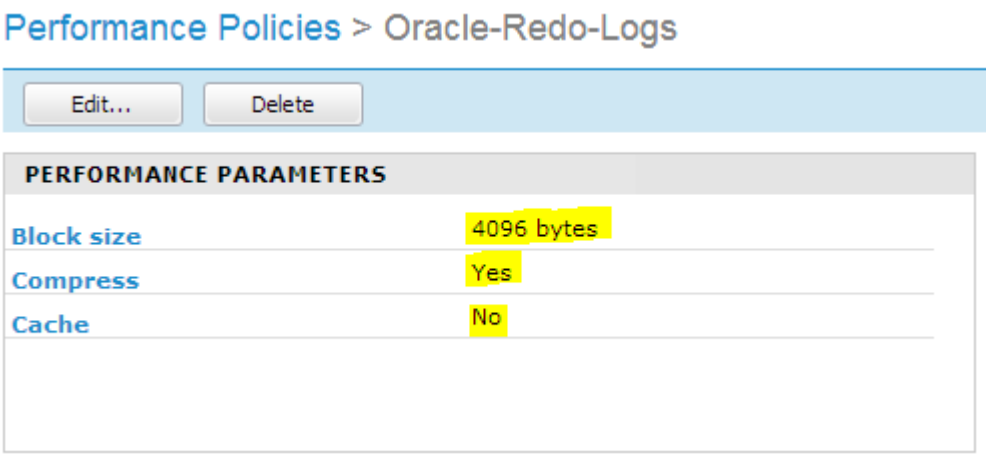
Table 1: OLTP Workload

File Type	# of Volumes	File System or ASM	Nimble Storage Caching Policy	Volume Block Size (Nimble Storage)	Oracle DB Block Size
Grid Infrastructure / Oracle Software	1	File System	Yes	8KB	N/A
OCR / Voting Disk	1	ASM	Yes	8KB	N/A
Database data files / Control files	8	ASM	Yes	8KB	8KB
Online Redo Logs	4	ASM	No	4KB	N/A
Archive Logs/ other logs	8	ASM	No	32KB	N/A

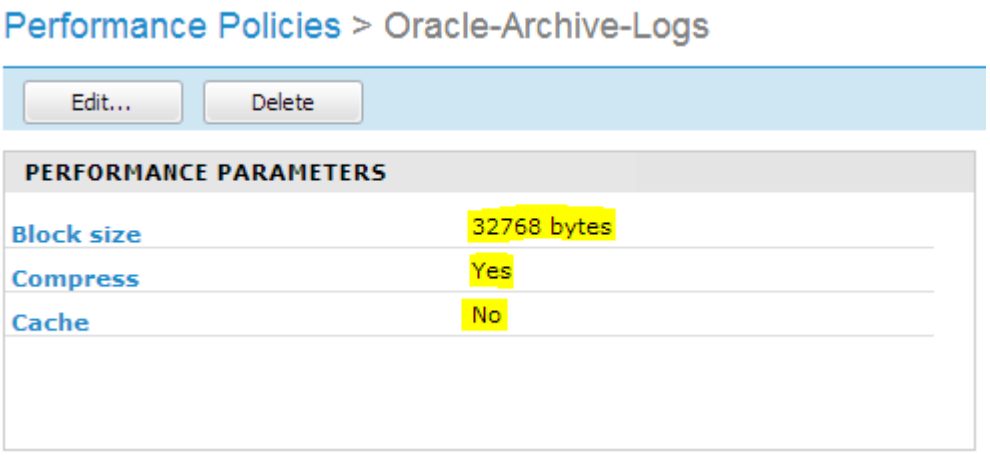
Here is the performance policy for Database data files/control files/GI & Oracle software/OCR & Voting disk.



Here is the performance policy for Online Redo Logs



Here is the performance policy for Archive Redo Logs



 As with any workload, be sure to evaluate the active data set and the cache hit rate through the Nimble Storage management tool to ensure optimal performance of the transactional database.

 Consider turning on aggressive caching when migrating databases from a legacy storage system. This can ensure faster response times as some of the active data may already be loaded into the flash cache.

Table 2: DSS Workload

File Type	# of Volumes	File System or ASM	Nimble Storage Caching Policy	Volume Block Size (Nimble Storage)	Oracle DB Block Size
Grid Infrastructure / Oracle Software	1	File System	Yes	8KB	N/A
OCR / Voting Disk	1	ASM	Yes	8KB	N/A
Database data files / Control files	8	ASM	Yes	32KB	32KB
Online Redo Logs	4	ASM	No	4KB	N/A
Archive Logs/ other logs	8	ASM	No	32KB	N/A

Using Snapshot and Zero-Copy Clones Features

Nimble Storage recommends using the native Nimble Storage snapshot feature to protect Oracle databases. Whether the databases are configured with Oracle Automatic Storage Management (ASM) or *nix File Systems, the snapshot feature can be invaluable. Below are some business requirements that can be achieved quickly and efficiently using snapshot.

- Whole database backup
- Database refreshes for Test/Dev/QA
- Offload RMAN backups to another server
- Database replication
- Test Oracle and Operating System patches

For more information on how to achieve these objectives, please see the “Nimble Storage Oracle Backup and Recovery guide” and the “Oracle Test and Development using Nimble Storage Zero-Copy Clones” technical reports.

Nimble Storage, Inc.

2740 Zanker Road., San Jose, CA 95134

Tel: 877-3NIMBLE (877-364-6253) | www.nimblestorage.com | info@nimblestorage.com

© 2013 Nimble Storage, Inc. Nimble Storage, InfoSight, and CASL are trademarks or registered trademarks of Nimble Storage, Inc. All other trademarks are the property of their respective owners. BPG-ORLX-0713