White Paper

ORACLE® BEST PRACTICES WITH XtremIO

Best Practices on Linux 6.x with EMC XtremIO Storage Array

Abstract

This white paper describes the best practices and recommendations to be adopted for the EMC XtremIO Storage Array when deploying an Oracle® Database Management System (DBMS) on Linux 6.x operating in a Fibre Channel SAN environment. It also explains how XtremIO's unique features (such as Inline Data Reduction techniques [including inline deduplication and data compression], scalable performance, data protection, etc.) provide high performance, simplicity and space saving benefits of deploying XtremIO as the primary storage for the Oracle Database.

March 2015

EMC²

Part Number H13497

Table of Contents

Executive Summary

Flash storage is an attractive method for boosting I/O performance in the data center. But it has always come at a price, both in high costs and loss of capabilities like scalability, high availability, and enterprise features.

XtremIO's 100% flash-based scale-out enterprise storage array delivers not only high levels of performance and scalability, but also brings new levels of ease-of-use to SAN storage, while offering advanced features that have never been possible before.

XtremIO's ground-up all-flash array design was created for maximum performance and consistent low latency response times, while also offering enterprise grade high availability features, real-time Inline Data Reduction that dramatically lowers costs, and advanced functions such as thin provisioning, tight integration to VMware®*, snapshots, volume clones, and superb data protection.

This white paper provides guidance and 'rules-of-thumb' methodologies that are recommended for deploying Oracle 11G R2 on Linux 6.x for physical servers in Fibre Channel SAN environments. This includes detailed descriptions of the components that are used to achieve optimum performance, and paramount levels of efficiency, when deploying Oracle Databases 11G R2 on XtremIO.

* VMware is a trademark or registered trademark of the VMware, Inc.

Introduction

Oracle's Database Management System (DBMS) operates at peak performance on the XtremIO Storage Array solution, irrespective of the workload it encounters, including running online transaction processing (OLTP), data warehousing and hybrid workloads.

The XtremIO Storage Array delivers predictable high performance and consistent low latency. The recommendations and best practices described in this white paper are geared to assist storage administrators to maximize the performance and data capacity utilization of the XtremIO Storage Array, when deploying an Oracle DBMS on Linux 6.x.

Scope

The scope of this white paper includes the following systems:

- Industry standard servers based on Intel x86 technology
- Linux 6.x:
  - Oracle Linux with Unbreakable Enterprise Kernel (UEK R1) 2.6.32-100.28.5.el6.x86_64 or higher
  - Oracle Linux with Unbreakable Enterprise Kernel (UEK R2) 2.6.39-100.5.1 or higher
  - Oracle Linux with the Red Hat Compatible Kernel
  - Red Hat Enterprise Linux 6.x (2.6.32-71.el6.x86_64 or later)
- Oracle Database 11G (11.2.0.3 and higher) Grid and Database Software
- Oracle ASM For Grid Infrastructure and Database
- Device Mapper For Multi-Path Software
- QLOGIC and Emulex HBAs

The following URL link provides additional detailed information; refer to "Certification Information for Oracle Database on Linux x86-64".

https://support.oracle.com/epmos/faces/DocumentDisplay?id=1304727.1&_adf.ctrl-state=5cmnuu2np_4&_afrLoop=249903916644922

LINUX 6.x on Physical Servers - Best Practices

Fibre Channel Switch Zoning Configuration

For zoning against XtremIO's Storage Array, it is recommended that at least two host bus adapters (HBAs) be available. Also, for improved performance, at least two active 8GFC ports should be configured per CPU socket. It is also recommended to zone initiators to all available storage ports. The recommended maximum number of paths to storage ports per host is 16, as described below in Table 1 through to Table 6.

To ensure a balanced utilization of XtremIO's full range of resources, it is recommended to utilize all storage ports in uniform, across all hosts and clusters.

In the tables below `XN_SCN_FCN` pertains to the Fibre Channel (FC) ports made available on each storage processor (XtremIO Storage Controller).

The XtremIO Storage Arrays currently support a maximum of six X-Bricks. Each X-Brick is comprised of two Storage Controllers, each of which has two FC ports designated respectively as FC1 and FC2.

Table 1. One X-Brick Two HBAs

2 HBAs	1 X-Brick		Ports Per Cluster
HBA1	X1_SC1_FC1	X1_SC2_FC1	4
HBA2	X1_SC1_FC2	X1_SC2_FC2	

Table 2. Two X-Bricks Two HBAs

2 HBAs	2 X-Bricks		Ports Per Cluster
HBA1	X1_SC1_FC1	X1_SC2_FC1	8
	X2_SC1_FC1	X2_SC2_FC1	
HBA2	X1_SC1_FC2	X1_SC2_FC2	
	X2_SC1_FC2	X2_SC2_FC2	

Table 3. Four X-Bricks Two HBAs

2 HBAs	4 X-Bricks		Ports per Cluster
HBA1	X1_SC1_FC1	X1_SC2_FC1	16
	X2_SC1_FC1	X2_SC2_FC1	
	X3_SC1_FC1	X3_SC2_FC1	
	X4_SC1_FC1	X4_SC2_FC1	
HBA2	X1_SC1_FC2	X1_SC2_FC2	
	X2_SC1_FC2	X2_SC2_FC2	
	X3_SC1_FC2	X3_SC2_FC2	
	X4_SC1_FC2	X4_SC2_FC2	

Table 4. One X-Brick Four HBAs

4 HBAs	1 X-Brick		Ports per Cluster
HBA1	X1_SC1_FC1	X1_SC2_FC1	8
HBA2	X1_SC1_FC2	X1_SC2_FC2	
HBA3	X1_SC1_FC1	X1_SC2_FC1	
HBA4	X1_SC1_FC2	X1_SC2_FC2	

Table 5. Two X-Bricks Four HBAs

4 HBAs	2 X-Bricks		Ports per Cluster
HBA1	X1_SC1_FC1	X1_SC2_FC1	16
	X2_SC1_FC1	X2_SC2_FC1	
HBA2	X1_SC1_FC2	X1_SC2_FC2	
	X2_SC1_FC2	X2_SC2_FC2	
HBA3	X1_SC1_FC1	X1_SC2_FC1	
	X2_SC1_FC1	X2_SC2_FC1	
HBA4	X1_SC1_FC2	X1_SC2_FC2	
	X2_SC1_FC2	X2_SC2_FC2	

Table 6. Four X-Bricks Four HBAs

4 HBAs	4 X-Bricks		Ports per Cluster
HBA1	X1_SC1_FC1	X1_SC2_FC1	16
	X2_SC1_FC1	X2_SC2_FC1	
HBA2	X1_SC1_FC2	X1_SC2_FC2	
	X2_SC1_FC2	X2_SC2_FC2	
HBA3	X3_SC1_FC1	X3_SC2_FC1	
	X4_SC1_FC1	X4_SC2_FC1	
HBA4	X3_SC1_FC2	X3_SC2_FC2	
	X4_SC1_FC2	X4_SC2_FC2	

Configuring HBA Settings

The following URLs are links to EMC documents that provide configuration details, including recommended settings:

- https://support.emc.com/docu6350_Host-Connectivity-with-QLogic-Fibre-Channel-and-iSCSI-Host-Bus-Adapters-(HBAs)-and-Converged-Network-Adapters-(CNAs)-in-the-Windows-Environment.pdf?language=en_US
- https://support.emc.com/docu6349_Host-Connectivity-with-Emulex-Fibre-Channel-Host-Bus-Adapters-(HBAs)-and-Fiber-Channel-over-Ethernet-Converged-Network-Adapters-(CNAs)-for-the-Linux-Environments.pdf?language=en_US

The only amendments to the above-referenced documents are the following:

- After connecting the first host to XtremIO, the recommended LUN queue depth setting is the maximum value that is supported per HBA (as outlined above):
  - 256 for a QLogic HBA
  - 28 for an Emulex HBA
- For connecting two hosts, the best practice is to reduce this setting by a half of the maximum value:
  - 128 for a QLogic HBA
  - 64 for an Emulex HBA

As the number of hosts that are connected to the array increases, it is recommended to decrease the LUN-queue depth setting proportionately. The absolute minimum setting, regardless of the number of hosts attached, for either QLogic or Emulex HBA type is 32, as shown in Table 7.

Table 7. Minimum HBA Settings (QLogic / Emulex)

Number of Hosts	Number of HBAs per Host	HBA Type	Number of HBAs per Host	LUN-Queue Depth Recommended Setting	Number of X-Bricks
1	2	QLogic	2	256	1-4
2	2	QLogic	2	128	1-4
3	2	QLogic	2	64	1-4
4+	2	QLogic	2	32	1-4
1	2	Emulex	2	128	1-4
2	2	Emulex	2	64	1-4
3+	2	Emulex	2	32	1-4

Mapping Volumes to the Initiator Group

Mapping XtremIO volumes to the Initiator Group via the XtremIO GUI, or via the XtremIO command line interface (XMCLI), ensures that the volumes are exposed and remain readily available to all of the paths that are provided. The host initiator ports, comprising of the Initiator Group zoned to all of the available storage ports, determine which device paths are made available to the host.

Installing and Configuring the Device Mapper

The device mapper (also known as DM-MPIO) is a Linux multipathing software that is suitable for balancing I/O to XtremIO Storage Arrays.

For more background on DM-MPIO refer to My Oracle Support note 753050.

XtremIO provides vital product data (VPD) via the `/etc/multipath.conf` configuration file. This file updates the device mapper with the multipath policies used for employing an active/active storage disk array, including load balancing to the multibus specification standards across paths, failover in the event of path failures, and failback once the paths have become re-established.

To install and configure the device mapper:

1. Via YUM*, install the device mapper.
2. Via YUM, install the device mapper-multipath, as follows:
   a. Use `chkconfig` to enable `multipathd` daemon:
      `# chkconfig multipathd on`
   b. Verify that the `multipathd` daemon is active after boot:
      `chkconfig –list|grp multipathd`
   c. Enter the XtremIO related parameters to the configuration file: `/etc/multipath.conf`, as follows:

```
defaults {
  user_friendly_names yes
}
devices {
  device {
                vendor              XtremIO
                product             XtremApp
                path_grouping_policy      multibus
                path_selector             "queue-
length 0"
                rr_min_io_rq          1
          }
}
```

3. Restart multipathd daemon. As the root user, at the shell prompt, enter:
   `#service multipathd restart`

Note:
The `user_friendly_names` parameter in the above example is set to `yes`.
If `queue-length 0` is set for the `path_selector` parameter, the algorithm checks the amount of outstanding path I/Os in order to determine which path to use next.
On Linux 6.x, the `rr_min_io_rq` parameter defaults to the value 1. Therefore, there must be at least one I/O request in the current path before the next path is used.

* YUM (Yellowdog Updater, Modified): The open-source command-line package-management utility for Linux operating systems using the RPM Package Manager.

Ensuring LUN Accessibility

To ensure that XtremIO devices are properly exposed, and remain readily accessible via the host without requiring a reboot of the host, it is necessary to complete either one of the following procedures:

- Via YUM CLI, install the `sg3_utils` package.

  The `sg3_utils` parameter creates the `rescan_bus_scsi.sh` configuration file, which is used to enable rescanning the bus for new devices without rebooting the cluster.

  To probe the SCSI bus for new LUNs on channels, execute the following as root:

  ```
  # rescan-scsi-bus.sh
  ```

OR

- Create a rescan script:

  As root, enter the following parameters at the shell prompt:

  - `echo "- - -">/sys/class/scsi_host/host1/scan`
  - `echo "- - -">/sys/class/scsi_host/host2/scan`

Note:
The `host<N>` path element pertains to the HBA port that is made available on the host and zoned to the XtremIO Storage Array storage ports.

Ensuring the Device Mapper Updates with New Devices

To ensure that the device mapper's nodes (or the nodes of any other multipath pseudo devices, such as `mpathXX`) are regularly updated and exposed, it is necessary to complete either one of the following steps:

At the shell prompt, enter:

- `# service multipathd restart`

OR

- `# multipath –F”;”multipath –v2`

Verifying that Paths are Discovered

- At the shell prompt, enter: `# multipath –ll`

Sample Output of Multipath(8) Command:

Each multipath device is assigned to an associated unique worldwide identifier (WWID). Setting `user_friendly_names` to `yes` automatically creates a device alias. This alias is defined as `mpathd<N>`, which is then uniquely assigned to the host.

Setting the `user_friendly_names` parameter to `no` sets the unique WWID as the multipath device name. It is entirely up to the implementer to decide which path to take.

The recommended best practice is to access block devices via the `/dev/mapper` directory. The output `mpathdo` (shown in the example below) becomes the multipath device. Four paths are displayed to notify that zoning is active (as described in Fibre Channel Switch Zoning, on page 6).

The SCSI devices (sdb, sdk, sdt and sdac) indicate that there are four paths to the same XtremIO volume via two initiators, four storage ports, and a channel LUN ID.

Example:

```
# multipath -11
mpathdo (3514f0c56b8c00007) dm-2 XtremIO,XtremApp
size=500G features='1 queue_if_no_path' hwhandler='0'
wp=rw
`-+- policy='queue-length 0' prio=1 status=active
  |- 1:0:2:1  sdb  8:16   active ready running
  |- 1:0:8:1  sdk  8:160  active ready running
  |- 2:0:0:1  sdt  65:48  active ready running
  `- 2:0:6:1  sdac 65:192 active ready running
```

Identifying XtremIO Volume Mappings to the Host LUN

A special mechanism is required in order to identify XtremIO volumes presented as LUNS to the Linux host. This mechanism is provided by the network address authority (NAA). This information is accessible via XMCLI commands, and clearly identifiable via the property window of any given volume.

To identify XtremIO volumes presented as LUNS:

1. From the XtremIO GUI menu bar, access the **Configuration** workspace, by clicking the **Configuration** icon.
2. In the **Volumes** pane, click a volume.

   In the **Volume Properties** window, note the unique NAA Identifier field (framed in red in Figure 1), identifying the volume uniquely presented to the host.

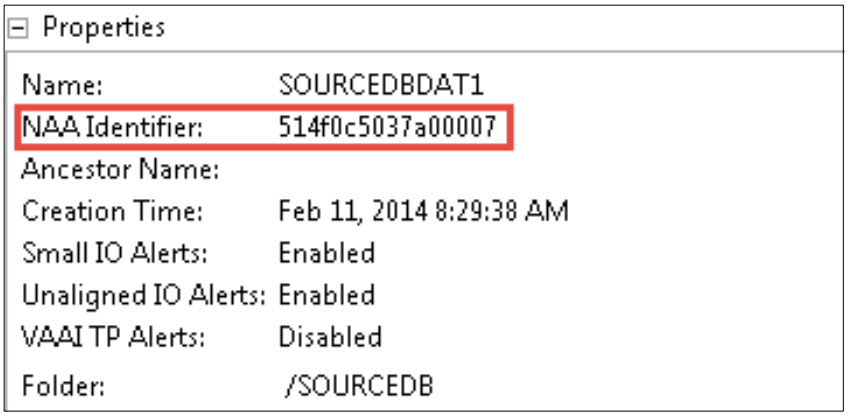

Figure 1. Volume Properties NAA Identifier Field

3. At the XMCLI prompt, enter `show-volumes vol-id= <volume name>` for each pertinent volume.

Table 8 shows volume names and the 12th field, indicating the NAA.

Table 8. Sample NAAs

Volume-Name	NAA (12th Field)
OEL63_DATA1	514f0c56b8c00007
EL63_DATA2	514f0c56b8c00008
OEL63_DATA3	514f0c56b8c00009
OEL63_DATA4	514f0c56b8c0000a
OEL63_DATA5	514f0c56b8c0000b
OEL63_DATA6	514f0c56b8c0000c
OEL63_DATA7	514f0c56b8c0000d
OEL63_DATA8	514f0c56b8c0000e
OEL63_REDO1	514f0c56b8c0001c
OEL63_REDO2	514f0c56b8c0001d
OEL63_REDO3	514f0c56b8c0001e
OEL63_REDO4	514f0c56b8c0001e

"3514f0c56b8c00007" (shown in the previous multipath example) is the WWID as detected by the host perspective, whereas "514f0c56b8c00007" is the NAA reported by XtremIO Storage Array.

Oracle ASM

Oracle Automatic Storage Management (ASM) is Oracle's recommended software for containing Oracle database files.

ASM Redundancy

External redundancy is generally recommended for XtremIO. The XtremIO Storage Array natively provides flash-optimized data protection.

ASM Allocation Unit Size

The ASM default allocation unit sizes are suitable for use with XtremIO.

Number of Disk Groups

The best practices for grouping Oracle DBMS file types in ASM disk groups are outlined in Table 9 and Table 10.

Table 9. Single-Instance Database

Database Type	DataDG	Redo<X>DG	Redo<Y>DG	FRADG
• Single-Instance	• First Control File • SPFILE • Data Files, Temp, Undo	• Redo Logs	• Multiplexed Redo Logs (if applicable)	• Archive Logs • Flashback Logs • Second Control File • Backup Components

Note:
The second redo data group (DG) is applicable if redo logs are multiplexed.

Table 10. RAC Database

Database Type	GridDG	DataDG	Redo<X>DG	Redo<Y>DG	FRADG
• RAC	• OCR • Voting File • SPFILE	• First Control File • Data Files, Temp, Undo	• Redo Logs	• Multiplexed Redo Logs	• Archive Logs • Flashback Logs • Second Control File • Backup Components

Note:
The second redo data group (DG) is applicable if redo logs are multiplexed.

Number of LUNS per Disk Group

Excellent cluster performance is achieved using an XtremIO Storage Array with just a single LUN in a single disk group. However, in order to maximize performance from a single host, adequate utilization of device queues and parallelism are required.

The best practice to achieve this is using four LUNS for the data disk group per array. Doing so enables the hosts, or applications, to use parallelism at various queuing points. This method ensures optimal performance from the XtremIO Storage Array.

The best practices for disk group configuration and data placement are described in Table 11 and Table 12.

Table 11. Single-Instance Database

Database Type	GridDG	DataDG	Redo<X>DG	Redo<Y>DG
Single-Instance	4 LUNs per Array	1 LUN	1 LUN	1 LUN per component – archive, flashback, backup, etc.

Table 12. RAC Database

Database Type	GridDG	DataDG	Redo<X>DG	Redo<Y>DG	FRADG
RAC	1 LUN	4 LUNs per Array	1 LUN	1 LUN	1 LUN per component – archive, flashback, backup, etc.

Creating a Linux Partition as Required by ASMLib

Oracle ASMLib is an optional host software that offers another method for handling persistent device naming and other features also found generically in later releases of Linux. Although many DBAs prefer Linux udev(8) for device naming, some may still want to use ASMLib. The URL below links to Oracle documentation that covers the options and combinations of ASMLib, ASM Filter Driver and udev(8).

http://docs.oracle.com/database/121/OSTMG/asminst.htm#OSTMG95908

For DBAs wishing to transition away from ASMLib, My Oracle Support note 1461321.1 details a step-by-step guide for converting from ASMLib to udev(8).

If ASMLib is required for specific business needs, the following information should be considered.

On various Linux releases, a partition is required when using ASMLib. Many utilities may be used to create partitions such as parted(8) or fdisk(8). Since fdisk(8) cannot function with very large devices, the choice of utility is usually influenced by the size of the device.

The first addressable sector for each device is sector 0, and each sector is 512 bytes in size. As a general rule, the best practice when partitioning the device is to explicitly assign the starting offset, such as one megabyte. This one megabyte of extra room is reserved by defining the partition to start at sector 2048. The extra room is available for storing the ASMLib header which serves to minimize the occurrence of ASMLib header corruption.

Note:
As recommended, partitioning drives also guarantees that I/O requests will be aligned properly for XtremIO.

Example for Using an fdisk

1. At the shell prompt, enter:
   ```
   # fdisk –u /dev/mapper/<mpath<NN>
   ```
2. Enter the following values:
   - n – for new
   - p – for partition
   - 1 – for partition 1
   - 2048 – for the starting sector
   - Enter – to accept the last sector
   - w – to save

To Access the recently created partition on the block device:

- Use the kpartx(8) command, such as the following example syntax:
  ```
  # kpartx –av /dev/mapper/mpathdN
  ```
  The addressable block device partition becomes `/dev/mapper/mpathNp1`.

If `/dev/mapper/mpathdNp1` is not displayed, it is necessary to restart `multipathd` via the service(8) command such as:

```
# service multipathd restart
```

Alternatively, you can use the multipath(8) command such as:

```
# multipath –F ; multipath –v2
```

Example for Initializing the LUN for Oracle ASMLib

At the shell prompt execute the oracleasm command, such as the following example syntax:

```
# oracleasm createdisk OEL63_DATADG_DISK1
/dev/mapper/mpath<NN>p1
```

In Linux clustering, it is common for hosts to assign different "friendly names" (e.g. mpathX) to share LUNs when the hosts boot up. This is often referred to as "device slip". Device slips can be prevented with udev(8). However, since the topic at hand is ASMLib, it should be noted that the `oracleasm-support` package labels disks with cluster-wide unique headers on each device.

Enabling Load Balancing when Using ASMLib

In order to ensure that DM-MPIO nodes are suitably utilized for load balancing, it is recommended to explicitly modify the ASMLib configuration file. The best practice is to perform the modifications while the existing ASM disk groups are unmounted.

Modifying/etc/sysconfig/oracleasm

Example:

```
ORACLEASM_ENABLED=true

# ORACLEASM_UID: The default user owning the /dev/oracleasm
mount point

ORACLEASM_UID=oracle

# ORACLEASM_GID: The default group owning the /dev/oracleasm
mount point

ORACLEASM_GID=dba

# ORACLEASM_SCANBOOT: When set as "true", the system scans
for ASM disks upon boot

ORACLEASM_SCANBOOT=true

# ORACLEASM_SCANORDER: Matching patterns to order disk
scanning ORACLEASM_SCANORDER="dm"

# ORACLEASM_SCANEXCLUDE: Matching patterns to exclude disks
from scan ORACLEASM_SCANEXCLUDE="sd"
```

Recycling ASM Daemon to Effect Changes

At the shell prompt, enter:

```
/etc/init.d/oracleasm stop

/etc/init.d/oracleasm start
```

512e Versus Advanced Format or 4K Advanced Format Considerations

The default setting for XtremIO volumes is 512e. It is recommended not to alter this in order to use 4K Advanced Format for Oracle Database deployments. There are no performance ramifications when using 512e volumes in conjunction with an Oracle database. On the contrary, 4K Advanced Format is rejected by many elements of the Oracle and Linux operating system stack.

Many software components in both Oracle and Linux operating system layers do not function properly with 4096B logical sector sizes (also known as 4K Advanced Format). An example of Linux operating system functionality which is lost when choosing 4K Advanced Format is when using direct I/O (O_DIRECT) support on both EXT4 and XFS file systems.

Table 13 and Table 14 assist in visualizing the complexities of involving 4K Advanced Format drives in an ASM environment. Note how many object types there are and how only 512e is a simple, comprehensive choice. For this reason, it is recommended not to override the 512e default format provided by XtremIO.

Table 13. Single-Instance Database

Database Type	DataDG	RedoDG	Redo<X>DG	FRADG
Single-Instance	512e/4K	512e/4K	512e/4K	512e/4K

Table 14. RAC Database

Database Type	GridDG	DataDG	Redo<X>DG	Redo<Y>DG	FRADG
RAC	512e	512e/4K	512e/4K	512e/4K	512e/4K

My Oracle Support note 1630790.1 is the master note covering issues that stem from choosing 4K Advanced Format.

Multiblock I/O Request Sizes

Oracle Database performs I/O on data files in multiples of the database block size (db_block_size), which is 8KB by default. The default Oracle Database block size is optimal on XtremIO. XtremIO supports larger block sizes as well. In the case of multiblock I/O (e.g., table/index scans with access method full), one should tune the Oracle Database initialization parameter `db_file_multiblock_read_count` to limit the requests to 128KB. Therefore, the formula for `db_file_multiblock_read_count` is:

```
db_file_multiblock_read_count = 128KB / db_block_size
```

Historically, Oracle Database was optimized to perform very large transfers in order to mitigate the cost of seek suffered by multiblock reads on mechanical drives. In a seek-free environment, such as XtremIO, there is no need for such mitigation. Also, most modern Fibre Channel host bus adapters require Linux to segment large requests into multiple requests. For example, an application I/O request of one megabyte is fragmented by the Linux block I/O layer into two 512KB transfers in order to suit the HBA maximum transfer size.

Redo Log Block Size

The default block size for redo logs is 512 bytes. I/O requests sent to the redo log files are in increments of the redo block size. This is the blocking factor Oracle uses within redo log files and has nothing to do with the on-disk format of the XtremIO LUN.

The recommendation on XtremIO is to create redo logs with 4KB block sizes as per My Oracle Support notes 1681266.1 and 1918508.1

To create a redo log with 4K-block size, set the `_disk_sector_size_`override parameter to `TRUE` in the database instance.

Note:
Do not set this parameter in the ASM instance. Once the instance is running with `_disk_sector_size_override` set to `TRUE`, simply add more redo logs with the BLOCKSIZE option set to 4096 and then drop any redo logs that have the default 512B block size.

Grid Infrastructure Files – OCR/Voting

The block size for both Oracle Cluster Registry (OCR) and Cluster Synchronization Services (CSS) voting files are 512 bytes. I/O operations to these file objects are therefore sized as a multiple of 512 bytes.

This is of no consequence since the best practice with XtremIO is to create volumes with 512e formatting.

Performance Monitoring

XMCLI commands for the XtremIO Storage Array are documented in the XtremIO Storage Array User Guide.

The show-‹object›-performance command provides pertinent counters, such as read IOPS, write IOPS, IOPS (total), read bandwidth (MB/s), write bandwidth (MB/s), and bandwidth (total) over durations and frequencies. Objects with counters include target ports, initiator ports, Initiator Groups, the cluster, volumes, snapshots, latency and many other object and entities. The monitoring commands provide a perspective on balanced utilization of the Storage Controllers' storage ports, initiators, Initiator Groups, I/O data-path modules, etc.

New "folder" objects have been recently introduced in order to provide counters for groupings of Initiator Groups and volumes. These counters are especially useful for identifying performance metrics between applications, utilizing the XtremIO Storage Array and volume groups within applications or hosts.

Most notably, the `show-xenvs` command reports on the utilization of I/O data-path modules over a specified interval, on the various Storage Controllers comprising the XtremIO Storage Array.

Performance Monitoring Examples:

Figure 2 shows a screenshot of the XtremIO GUI dashboard with real-time monitoring of storage processor CPU on a dual X-Brick array.

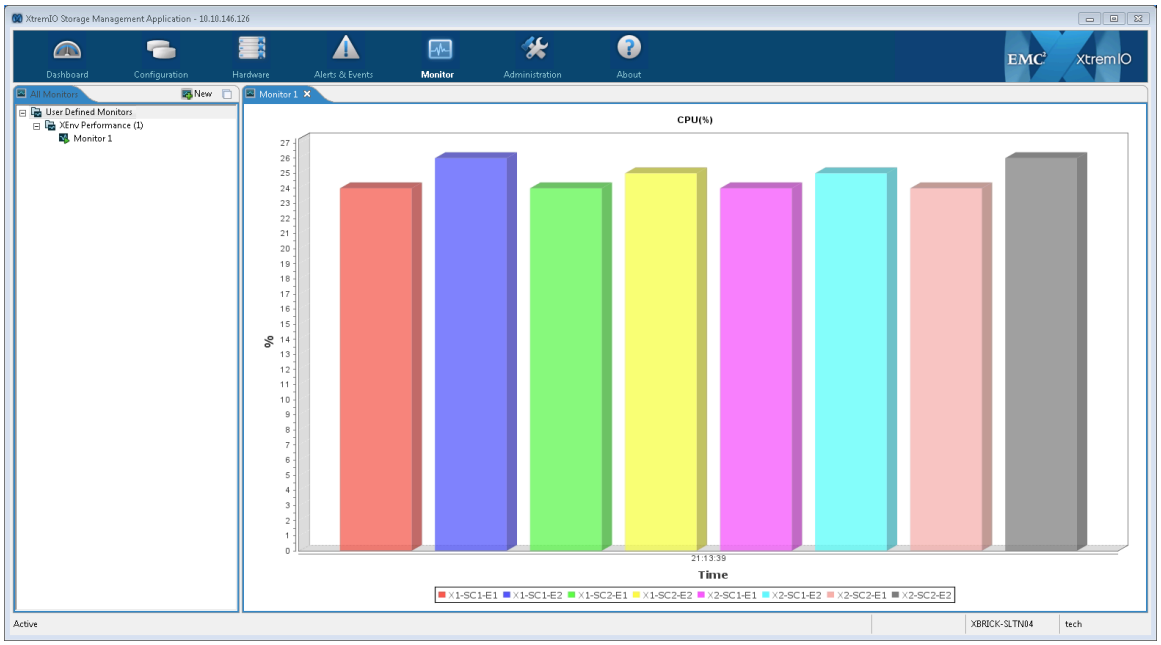

Figure 2. Dashboard with Real-Time Monitoring of Storage Processor CPU

Figure 3 shows initiator's connectivity to FC-target ports from SAN clients. The output should reflect balanced zoning implementation.

```
Connect XMS on 10.10.169.69:443: version 2.4.0 build 25
xmcli (tech)> show-initiators-connectivity target-details
Existing Initiator List:
Name           Index  Port-Type  Port-Address              Num-Of-Conn-Targets  Target-Name   Index
xioracc_host1  1      fc         21:00:00:24:ff:2f:ac:d8   2                    X1-SC1-fc1    1
                                                                               X1-SC2-fc1    11
xioracc_host2  2      fc         21:00:00:24:ff:2f:ac:d9   2                    X1-SC1-fc2    2
                                                                               X1-SC2-fc2    12
xioracd_host1  3      fc         21:00:00:24:ff:2f:ae:02   2                    X1-SC1-fc1    1
                                                                               X1-SC2-fc1    11
xioracd_host2  4      fc         21:00:00:24:ff:2f:ae:03   2                    X1-SC1-fc2    2
                                                                               X1-SC2-fc2    12
```

Figure 3. "show-initiators-connectivity target-details"

Note:
<HOSTNAME_hostX> designation pertains to HBA port discovered by the host.

Figure 4 shows balanced performance utilization of the FC-target ports from SAN clients.

```
Name          Index  Write-BW(MB/s)  Write-IOPS  Read-BW(MB/s)  Read-IOPS  BW(MB/s)  IOPS   Total-Write-IOs  Total-Read-IOs
X1-SC1-fc1    1      20.669          1936        291.688        37336      312.357   39272  885818433        2322777898
X1-SC1-fc2    2      21.136          1962        291.079        37258      312.215   39220  886111301        2323382567
X1-SC1-iscsi1 5      0.000           0           0.000          0          0.000     0      0                0
X1-SC1-iscsi2 6      0.000           0           0.000          0          0.000     0      0                0
X1-SC2-fc1    11     18.626          1919        289.683        37080      308.309   38999  888409698        2314638254
X1-SC2-fc2    12     18.537          1954        289.016        36994      307.553   38948  888466019        2315731056
X1-SC2-iscsi1 15     0.000           0           0.000          0          0.000     0      0                0
X1-SC2-iscsi2 16     0.000           0           0.000          0          0.000     0      0                0
```

Figure 4. "show-targets-performance frequency=30"

Figure 5 shows balanced performance utilization of initiators by two Storage Controllers (xioraca, xioracb).

Initiator-Name	Index	Write-BW(MB/s)	Write-IOPS	Read-BW(MB/s)	Read-IOPS	BW(MB/s)	IOPS	Total-Write-IOs	Total-Read-IOs
xioracc_host1	1	0.586	52	0.607	46	1.193	98	1407591360	3501429162
xioracc_host2	2	0.604	53	0.686	52	1.290	105	1408090509	3502713190
xioracd_host1	3	0.006	1	0.004	0	0.010	1	378667988	1369919257
xioracd_host2	4	0.004	0	0.013	2	0.017	2	378528046	1370343808

Figure 5. "show-initiators-performance frequency=30"

Figure 6 shows the comparative performance utilization of a group of application volumes on the XtremIO Storage Array.

Name	Index	Write-BW(MB/s)	Write-IOPS	Read-BW(MB/s)	Read-IOPS	BW(MB/s)	IOPS	Total-Write-IOs	Total-Read-IOs
/	1	3.915	434	1321.481	169148	1325.396	169582	682510253	1480770254
/11GR2-xioracc	2	3.884	434	1321.481	169148	1325.365	169582	132873227	85850594
/11GR2D-xioracd	3	0.031	0	0.000	0	0.031	0	549637026	1394919660
/snapcdb-xioracd	4	0.000	0	0.000	0	0.000	0	0	0
/snapddb-xioracc	5	0.000	0	0.000	0	0.000	0	0	0
/Test	6	0.000	0	0.000	0	0.000	0	0	0

Figure 6. "show-volume-folders-performance frequency=30"

Figure 7 shows the utilization of `I/O data_path` modules residing on various Storage Controllers.

XEnv-Name	Index	CPU(%)	CSID	State	Storage-Controller-Name	Index	Brick-Name	Index
X1-SC1-D	2	80	11	active	X1-SC1	1	X1	1
X1-SC1-RC	1	70	10	active	X1-SC1	1	X1	1
X1-SC2-D	4	85	13	active	X1-SC2	2	X1	1
X1-SC2-RC	3	71	12	active	X1-SC2	2	X1	1

Figure 7. "show-xenvs frequency=30"

Implementing Host Quality of Service (QoS)

The XtremIO Storage Array is an "equal-opportunity" array, servicing all I/O requests from all hosts with simple first-in-first-out fairness. Potentially non-mission-critical applications may utilize a larger share of the array's performance capacity than desired by the administrator. However, host I/O on Linux platforms can be easily managed with the Linux Control Groups.

The following references offer more information regarding implementing QoS at the host level:

- http://www.oracle.com/technetwork/articles/servers-storage-admin/resource-controllers-linux-1506602.html
- https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Resource_Management_Guide/ch-Subsystems_and_Tunable_Parameters.html#blkio-throttling

Example of Implementing Host QoS to Limit Performance of DEV Server:

At the shell prompt, enter:

```
mkdir /cgroup/blkio
mount -t cgroup -o blkio none /cgroup/blkio
cgcreate -t oracle:dba -a oracle:dba -g blkio:/iothrottle
```

Identifying the Device "Major:Minor" Number

```
[root@OEL63-1~]#ls-l/dev/oracleasm/disks/XIOLOCALDG
brw-rw---- 1 oracle dba 252, 52 Jun 13 14:22 /dev/oracleasm/disks/XIOLOCALDG
```

Limiting DEV Server Maximum Read IOPS to 10KB

```
[root@OEL63-1 ~]# echo "252:52 10000" > /cgroup/blkio/blkio.throttle.read_iops_device
```

Optionally Limiting Source Database Server Maximum Read IOPS to 100KB

```
[root@OEL63-1 ~]# echo "252:52 100000" > /cgroup/blkio/blkio.throttle.read_iops_device
```

Testing Benchmark Simulations on Source and Target Databases

Figure 8 and Figure 9 show the primary database server approaching 100,000 IOPS, whereas the development server (DEV) shows a much-reduced level of total IOPS; approximately 15KB IOPS.

Note:
In the examples below, only the reads were limited on the DEV server. The mix workload enables the total IOPS to exceed 10,000.

The following example is based on SLOB (xtremio.com/slob):

- ./runit.sh 64 – Source Database
- ./runit.sh 64 – Target Database

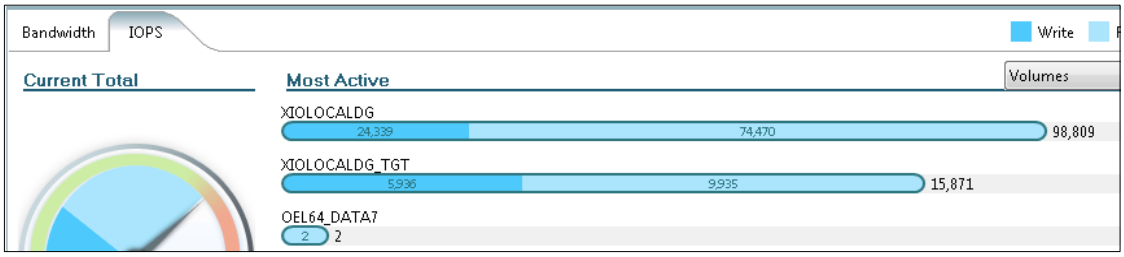

Figure 8. IOPS

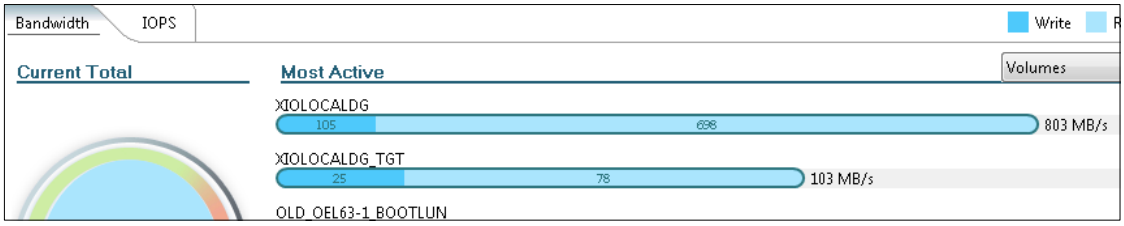

Figure 9. Bandwidth

Simplicity of Operation – Provision Capacity without Complexity

"Capacity" is the single concern for provisioning storage from XtremIO for Oracle Databases.

XtremIO LUN provisioning and presentation is very simple and can be performed via the XtremIO GUI, or the XMCLI. Provisioning storage to a host from XtremIO consists of the following steps.

1. On the XtremIO Storage Array:
    a. Create Volumes.
    b. Create an Initiator Group.
    c. Map the Volumes to the Initiator Group.
2. On the host
    a. Perform a "host LUN discovery".

Utilities for Thin Provisioning Space Reclamation

Oracle Automated Storage Management does not trim the space made potentially available by files that were deleted in the disk group. Instead of trimming space, ASM marks the free spaces as "available for overwrite", resulting in the XtremIO array reporting on logical used space inaccurately.

The ASM Storage Reclamation Utility® (ASRU)* injects released space with deleted files in an ASM disk group, with zero-byte blocks.

Executing ASRU against an ASM disk group that has had many deleted files serves to adjust accounting of logical used capacity on XtremIO. If deleted files are not referenced anywhere else, ASRU also corrects the reported physical used capacity.

Using an XFS and ext4† file systems, deleted files are automatically trimmed, by specifying the "Discard" mount option. These instructions then propagate to the array. Alternatively, one can forego the "Discard" mount option and perform trim operations out-of-band with the `fstrim(8)` command.

* ASRU is a trademark or registered trademark of the Oracle Corporation.
† The ext4 or "fourth extended filesystem" is a journaling file system for Linux, developed as the successor to ext3.

Snapshots Used for Backup-to-Disk

XtremIO Storage Array snapshots are precise point-in-time copies of source volumes which, essentially, are a collection of pointers referencing the source volume blocks. Therefore, snapshots consume no physical capacity.

Executing snapshots is an extremely rapid and efficient backup-to-disk methodology. This is because snapshots solely utilize metadata operations. As with source volumes, snapshots employ the same core benefits that are innately attributed to source volumes, including high-level performance, XtremIO Data Protection (XDP), automatic data distribution, global deduplication and thin provisioning.

The space that is saved by creating snapshots is not reflected in the deduplication ratio (as is the case with RMAN Image copies). This is because snapshots are pointer-based, and therefore, XtremIO snapshots are not actual duplicated blocks performed by a host process.

Instead, the space that is saved by snapshots is tremendous, which is especially the case directly after a snapshot creation process. Over time, as source volumes are updated (depending on the change rate), and snapshots are mounted and accessed for writes (or not), the net physical capacity consumed by both source volumes and snapshots grows. For backup purposes, it is imperative to invoke snapshots while the database is in "Backup" mode. This is done in order to create valid image copies on snapshots, and enables rolling-forward the database utilizing logs (such as offline logs and/or online or redo logs) up to a desired System Change Number (SCN). This establishes a consistent point in time or "latest SCN".

Invoking snapshots to roll-forward to a latest SCN establishes a latest consistent state (from a database perspective). As a minimum precaution, the recommended best practice is to create a backup control file prior to initiating a backup-to-disk process.

For recovery purposes, a recommended best practice is to separate data files and logs (both offline and online), hence enabling a recovery from various points-in-time.

XtremIO backup-to-disk image creation (snapshots) is a seamless and fast process, and results in no perceived degradation in terms of performance against the source volumes. Freeze and thaw of the source volumes are implicitly performed internally on XtremIO, via SCST during snapshot operations.

The Snapshot Groups* feature is supported to ensure that headers that reside in among the database files (such as control files, data files, log files and optional application volume[s]) remain consistent. Multiple snapshots or Snapshot Groups of the source volumes, as well as snapshots of snapshots, are fully supported.

* Snapshot Group refers to any snapshot action that is performed on a folder, or on a manually-selected list of volumes.

This support is provided to enable precautions to be made prior to attempting actual restores and recoveries, such as performing a mock restore and recovery.

Unlike the RMAN "Restore" process, the snapshot process for restoring is very fast. The current version of XtremIO does not support "Snapshot Restore to the Source Volume". Therefore, the "Restore" process is comprised of manually unmapping source volumes, mapping the snapshots and/or Snapshot Groups, and performing Oracle media recovery. However, this changes in version XIOS 4.0.

The best practice is to backup image copies on snapshots to another storage or tape.

Note:
Image copies on snapshots can be entered into RMAN repository metadata for inspection, by using the "`CATALOG`" command. Hence, RMAN may be used to back up the files to tape (or by other storage media) for long-term retention. Doing this ensures that the XtremIO primary database is not compromised, in the unlikely event of a total disaster involving the XtremIO Storage Array.

The following URL links to comprehensive details for RMAN backup concepts.

http://docs.oracle.com/cd/E11882_01/backup.112/e10642/toc.htm

The following URL links to comprehensive steps for backing up existing image copy backups with RMAN.

http://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmbckba.htm#BRADV89561

Snapshots Used for Manual Continuous Data Protection (CDP)

As the implementation of snapshots is so efficient on the XtremIO Storage Array, the snapshots feature may, arguably, be used for business continuance strategy or for continuous data protection (CDP). Two iterations can be used for this strategy:

- A crash-consistent, or "restartable" image

OR

- A recoverable image

Crash-Consistent Image

In basic terms, a crash-consistent or restartable image is a point-in-time image of the primary database on disk; a snapshot.

This iteration entails taking snapshots and/or Snapshot Groups of the primary database while it is up and operational. The image that is captured is similar to the state of the primary database, once the `shutdown abort` command is issued against it.

During the database restart on the snapshots and/or Snapshot Groups, the database automatically performs a recovery, using the online logs. All committed transactions are reflected and all uncommitted transactions are rolled back. The Recovery Point Objective (RPO) is defined per interval. The interval is defined by how often snapshot and/or Snapshot Group creation is scheduled, which can be set as daily, hourly, or defined in minutes (for example, every 30 minutes).

To perform a restore operation, unmount the disk groups or file systems (if applicable), and unmap all of the source volumes comprising the database (data files plus control file, online logs, archived log destination). Once these actions have been successfully performed, map the corresponding snapshot or Snapshot Group.

To perform a recover operation using SQLPLUS, at the prompt, enter `startup` against the snapshots primary database.

Recoverable Image

A recoverable image is an image of the primary database on disk; a snapshot. This iteration entails taking snapshots and/or Snapshot Groups of the primary database while the database is in "Backup" mode.

The image should be captured after the `alter database begin backup` command is issued against it. The `alter database end backup` command should be executed shortly thereafter, which is done in order to avoid excessive logging.

It is also highly recommended to have a backup file of the control file prior to commencing the backup process, and for after the completion of the backup process.

The recovery point objective (RPO) is defined per interval, once snapshots and/or Snapshot Groups are created. The interval can be set as daily, hourly, or defined in minutes.

Unlike with the crash-consistent image iteration, data files on the replica can be rolled forward through time. This is performed by using logs up to a consistent point in time, either to the desired SCN or up to the latest SCN (captured in the control file). This means that RPO is ultimately far more granular than it is with scheduled intervals. In this way, not only can an image be recovered via the scheduled intervals, but points in time in-between intervals can also be recovered. This works in conjunction with an Oracle media recovery.

Snapshots for Cloning Primary Databases

Clones of the primary database may be deployed, which is done by using the methodologies described above.

Oracle provides a utility called "NID" (or "NEWDBID®")* which is used to facilitate renaming of the `database ID` and database `Name` properties automatically, as opposed to having to recreate the control file.

* NID and NEWDBID are trademarks or registered trademarks of the Oracle Corporation.

Recovery Manager Image Copies for Backup to Disk

The Oracle Recovery Manager (RMAN) is an Oracle-native tool for backing up, restoring and recovering an Oracle database. The tool is an integral part of Maximum Availability Architecture (MAA) employed by Oracle for making robust database deployments.

RMAN creates backups on disk by default and also creates backup sets by default, rather than creating image copies. A backup set consists of physical files which can be written to either disk or tape, yet the format is native to RMAN only (as opposed to image copies).

Image copies created via the "`BACKUP AS COPY`" command are bit-by-bit copies of database files. Image copies may be directed to disk just like a backup set. The copies are then recorded in the RMAN repository, either via an RMAN catalog or via a control file of the target database (in cases where a catalog does not exist, or was not used).

Image copy is the recommended format for backup-to-disk, as the format provides the highest level of space savings on the XtremIO Storage Array. This is done by providing up to a 2:1 deduplication ratio between the source database and the RMAN backup to disk. Using RMAN to clone the primary database also gains benefits from XtremIO's deduplication feature. In an environment where the primary database, RMAN backup-to-disk (image copies), and clone of the primary database are all residing on the XtremIO Storage Array, the deduplication feature potentially boosts the benefit up to a 3:1 ratio.

Use Cases for RMAN Image Copies:

- Image copies may be used to restore control files, data files and logs, when primary files are corrupted or are inadvertently deleted.
- Image copies on disk may also be used as point-in-time copies of the actual database files. Thus, the time-consuming restore from the backup location to the actual primary volumes can be avoided. Irrespective of whether the image copies reside on ASM or on the file system, RMAN automatically re-directs the pointers to the image copies, updating the control files accordingly.
- Image copies on disk may also be used to create a clone of the primary database on the same host or on another host.
- Image copies may be used to create secondary, backup copies, either to tape media or to another storage device.

Appendix A – ASM Disk Group Sector Size – ASMLib Ramifications

512e is the prescribed best practice for XtremIO in conjunction with Oracle Database. This section is provided for informational purposes only as it pertains to the use of 4K Advanced Format in an Oracle Database deployment that is not compliant with XtremIO best practices.

The minimum I/O-transfer size for files in an ASM disk group is determined by the sector size of the underlying physical drive.

Oracle ASM queries devices for the logical sector size of the drive and assigns this value to the `sector_size` disk group attribute (see My Oracle Support note 1938112.1). This is the expected behavior for ASM disks that are not accessed with ASMLib. However, an exception to this behavior was exhibited in early versions of Linux 6.x, with native multi-pathing software (e.g. Device Mapper). In these older Linux versions, the physical sector size was adopted by ASMLib for the ASM disk group instead of the logical sector size.

When using EMC PowerPath, instead of Device-Mapper, Oracle queries the device and finds that the logical-sector size of the LUN is the same as the physical-sector size. Therefore, no work-around is required with ASMLib.

If your business requirements insist on the specific combination of 4K Advanced Format, ASMLib with DM-MPIO, and neither udev(8) control nor EMC Powerpath on XIOS 2.4, refer to My Oracle Support note 1526096.1 for more information on this matter.