Abstract

This white paper describes the deployment of the XtremIO® all-flash array with Oracle RAC 11g and 12c databases in both physical and virtual environments. It describes optimal performance while scaling up in a physical environment, the effect of adding multiple virtualized database environments, and the impact of using XtremIO Compression with Oracle Advanced Compression. The white paper also demonstrates the physical space efficiency and low performance impact of XtremIO snapshots.

May 2015
Table of contents

Executive summary.......................................................................................................................... 5
  Business case ................................................................................................................................. 5
  Solution overview ......................................................................................................................... 5

Introduction ..................................................................................................................................... 7
  Document purpose ......................................................................................................................... 7
  Scope ............................................................................................................................................ 7
  Audience ....................................................................................................................................... 7

Solution architecture ..................................................................................................................... 8
  Overview ....................................................................................................................................... 8
  Solution architecture ..................................................................................................................... 10
  Hardware resources ...................................................................................................................... 11
  Software resources ....................................................................................................................... 11

Storage layer: EMC XtremIO 3.0 .................................................................................................... 12
  XtremIO features .......................................................................................................................... 12
  Storage configuration .................................................................................................................... 12
  Storage design .............................................................................................................................. 13

Virtualization layer: ESXi and virtual machine .............................................................................. 14

Oracle Database ............................................................................................................................ 15
  Overview ....................................................................................................................................... 15

Use case 1: Scale out and consolidate production and non-production environments .............. 16
  Description ..................................................................................................................................... 16
  Configuration .................................................................................................................................. 16
  Testing detail .................................................................................................................................. 18
  Results ............................................................................................................................................ 18
    OLTP database workload performance metrics ........................................................................... 18
    Test results: Scaling the number of X-Bricks from one to four in the XtremIO array ................. 20
    Test results: Consolidating production and TEST/DEV database workloads on the four X-Brick
    XtremIO array ............................................................................................................................. 20
  Use case summary ....................................................................................................................... 21

Use case 2: Consolidate physical and virtualized databases onto the four-X-Brick XtremIO array ...... 22
  Description ..................................................................................................................................... 22
  Configuration .................................................................................................................................. 23
  Testing detail .................................................................................................................................. 25
  Results ............................................................................................................................................ 26
Use case 3: Validating how XtremIO compression works with Oracle Advanced Compression

Description .................................................................................................................. 30
Configuration .............................................................................................................. 31
Testing detail ................................................................................................................ 32
Results ......................................................................................................................... 33
  Relative physical storage space compression efficiency with and without Oracle ACO enabled on
  the XtremIO array ........................................................................................................ 33
  OLTP database workload performance with and without Oracle ACO enabled on XtremIO .... 35
Use case summary ........................................................................................................ 35

Use case 4: Validating the storage capacity and performance overhead of XtremIO snapshots

Description .................................................................................................................. 36
Configuration .............................................................................................................. 37
  Creating snapshots from the XtremIO Storage Management Application console .......... 37
  Creating snapshots from the AppSync console ........................................................... 39
  Mount the repurpose copy to the target server to provision TEST/DEV database .......... 42
Testing detail ................................................................................................................ 45
Results ......................................................................................................................... 46
Methodology ............................................................................................................... 46
  Test results: Performance impact of XtremIO snapshots on the production database .... 47
  Test results: Performance impact of the DEV workloads on the production workload ...... 51
  Test results: Creating cross-consistent snapshots for the volumes of the databases ...... 52
Use case summary ........................................................................................................ 54

Conclusion ................................................................................................................... 55
Summary ....................................................................................................................... 55
Findings ......................................................................................................................... 55

References .................................................................................................................... 57
  EMC documentation ................................................................................................... 57
  Oracle documentation ................................................................................................. 58
  VMware documentation ............................................................................................. 58

Appendix ...................................................................................................................... 59
  SLOB configuration parameters .................................................................................. 59
Executive summary

Business case

Business needs are driving data growth, in both volume and velocity, more than ever before. At the same time, the need to quickly convert the data into useful information that reveals business opportunities and risks is also growing. Relational databases, such as Oracle Database 12c, are used to support these business-critical applications. To deliver faster response times across the range of applications, these databases require storage designed for both low-latency transactional input/output (I/O) and high-throughput analytic workloads.

Virtualization enables greater consolidation of various types of database workloads. Often, because of consolidation, both online transaction processing (OLTP) and online analytical processing (OLAP) workloads share the same physical servers and storage. Therefore, to enable optimal performance, the underlying storage infrastructure must also be designed to handle these varying workloads within a consolidated infrastructure.

For database administrators, storage teams, and application owners optimizing end-to-end performance through complex database layout and configuration, Oracle databases consume too much time. This problem is more severe in mixed workload environments with OLTP, data warehouses, reports, analytics or test and development (test/dev). This problem is illustrated in the 2014 IOUG survey, “Efficiency Isn’t Enough,” which shows that 48 percent of the DBA’s weekly time is spent doing performance diagnostics on the database environments and 37 percent of the DBA’s weekly time is spent creating Oracle database copies.

The EMC® XtremIO™ all-flash array addresses the effects of virtualization on I/O-intensive database workloads with impressive random I/O performance and ultra-low latency. This applies equally to OLTP and OLAP workloads and to the consolidation of multiple workloads onto a common storage platform. XtremIO also provides new levels of speed and provisioning agility to virtualized environments with space-efficient snapshots, inline data reduction, and accelerated provisioning and compression. The results include breakthrough simplicity for storage management and provisioning, and new capabilities for real-time analytics and test/dev cycles.

Solution overview

This solution demonstrates the benefits of deploying different versions of Oracle databases in both physical and virtual environments using VMware vSphere and XtremIO 3.0 storage. Once Oracle databases are deployed in production, IT is required to copy instances for functions such as reporting, analytics, and test/dev. Traditional storage cannot deliver all these copies from a single platform with consistent performance and data reduction.

XtremIO creates easy, on-demand memory-based snapshots and VM clones that have no impact on production SLAs with the same performance as production, and are space-efficient for any new and unique data. XtremIO gives Oracle customers one platform with radical simplicity and no tuning for all workloads. Every database has all-flash arrays for predictable performance and low latency.
The benefits of the solution include:

- **Simplify database consolidation**—Consolidate mixed-workload, test/dev databases with performance at scale on a single platform. Response times remain consistently low in both physical and virtual environments.

- **Maximize DBA productivity**—Simplify provisioning of new Oracle databases. Reduce downtime for capacity planning & growth management. Accelerate Oracle test/dev with snapshots.

- **Gain greater business insight**—Run faster transactions. Increase queries and users, and improve business efficiency.
Introduction

**Document purpose**  This white paper highlights:

- EMC XtremIO performance optimization in both virtualized and physical Oracle RAC 11g and 12c
- Multiple physical and virtualized Oracle databases running Silly Little Oracle Benchmark (SLOB) random I/O workloads
- The impact of Oracle’s Advanced Compression Option (ACO) and XtremIO compression working together
- XtremIO snapshot capacity overhead, and its performance impact on a random I/O workload

**Note:** For more information about SLOB, refer to xtremio.com/slob and kevinclosson.net/slob

**Scope**  This white paper describes the solution architecture and the procedures that were used to validate the following use cases:

- Scale out and consolidation of both production and non-production environments
- Consolidation of physical and virtualized databases on the same XtremIO array
- Use of Oracle Advanced Compression with XtremIO compression
- File system capacity usage and I/O performance overhead of XtremIO snapshots

**Audience**  This paper is intended for Oracle database administrators, storage administrators, virtualization administrators, system administrators, IT managers, and any others involved in evaluating, acquiring, managing, maintaining, or operating Oracle database environments.
Solution architecture

Overview
In this solution, the scalability, versatility, compression characteristics, and snapshot efficiency of the XtremIO array with Oracle 11g and 12c databases are tested using the following use cases:

- **Scaling XtremIO from 1 to 4 X-Bricks.** We deployed a production physical Oracle 12c two-node RAC OLTP database on a single X-Brick XtremIO array, and populated 20 TB test data to the database using SLOB.

  We validated how the performance of the physical production database was affected as we scaled the number of XtremIO X-Bricks from one to four and quadrupled the workload at the same time in the following way:

  - A SLOB random I/O workload was run on the single X-Brick configuration XtremIO array to capture and validate the database performance baseline.
  - Next, the number of X-Bricks on the XtremIO array was scaled up from one to four and the SLOB workload was quadrupled.
  - The performance statistics were recorded on the four-X-Brick configuration and compared to the original single X-Brick performance baseline. This was to highlight the performance impact on the workload running on the Oracle RAC 12c Database as the XtremIO scales.

  To validate how the performance of the physical production database was impacted as the test/dev workloads were consolidated to the same array, we performed the following:

    - Two XtremIO cross-consistent snapshots of the production database volumes were created and mounted to two physical servers to provision single instance TEST and DEV databases.
    - Simulated test/dev workloads were run against the provisioned test/dev databases.
    - The performance impact of the test/dev workloads on the production database was validated.

  For a more detailed discussion of XtremIO cross-consistent snapshot, refer to the *Introduction to XtremIO Snapshots* white paper in the References section.

  See Use case 1: Scale out and consolidate production and non-production environments.

- **Consolidate physical and virtualized databases on the same array.** We recorded a performance baseline by running a SLOB workload to the physical production database. We deployed multiple Oracle databases on VMware vSphere virtual machines to simulate virtualized production Oracle databases. The same SLOB workload was run on the virtual databases to validate the performance impact on the workload that was running on the physical production database.

  We then provisioned a virtualized test/dev database based on the XtremIO snapshots of the volumes of the virtualized Oracle 12c database. Next, we ran a simulated test/dev workload on the virtualized test/dev database to validate
the performance impact on the workload running on the source virtualized production database

See Use case 2: Consolidate physical and virtualized databases onto the four-X-Brick XtremIO array.

- **Validating XtremIO compression with Oracle Advanced Compression.** We deployed a virtual machine from a template and created an Oracle 12c database. In the database, we created four 1 TB tablespaces on separate ASM diskgroups. The same Swingbench\(^1\) Order Entry schema data was loaded into each of the tablespaces.
  - The segments on the first tablespace were populated without Oracle Advanced Compression enabled, and the tables were created with PCTFREE=10.
  - The segments on the second tablespace were populated with Oracle Advanced Compression enabled, and the tables were created with PCTFREE=10.
  - The segments on the third tablespace were populated without Oracle Advanced Compression enabled, and the tables were created with PCTFREE=0.
  - The segments on the fourth tablespace were populated with Oracle Advanced Compression enabled, and the tables were created with PCTFREE=0.

We then validated the compatibility of XtremIO compression and Oracle Advanced Compression in terms of the physical capacity of the array and OLTP workload performance.

See Use case 3: Validating how XtremIO compression works with Oracle Advanced Compression

---

**Note:** PCTFREE is a storage parameter of the database table that is essential to how the database manages free space. This parameter sets the minimum percentage of a data block reserved as free space for updates to existing rows. Thus, PCTFREE is important for preventing row migration and for avoiding wasted space.

---

- **Validating the storage capacity and performance overhead of XtremIO snapshots.** We validated the performance impact of XtremIO snapshots on the random I/O workload by running SLOB on the production database before, during, and after creating XtremIO snapshots using EMC AppSync™. We also used AppSync to create 63 repurposed copies of the production database and provision 63 DEV databases based on the repurposed copies. To validate the performance impact of DEV workloads on the production workload, we ran simulated DEV workloads to these DEV databases while simultaneously running the production workload on the source database. Finally, we validated the array capacity overhead of XtremIO snapshots by creating multiple cross-consistent XtremIO snapshots of Oracle databases. We created Snapshots for Oracle databases with different versions, different volumes, and different sizes.

---

\(^1\) For more information about Swingbench, refer to [http://dominicgiles.com/swingbench.html](http://dominicgiles.com/swingbench.html).
EMC AppSync is advanced copy management software for EMC storage arrays that offers a better way to manage the protection, replication, and cloning of critical applications and databases. For more detailed information, please refer to the *EMC AppSync User and Administration Guide* in the References section.

See Use case 4: Validating the storage capacity and performance overhead of XtremIO snapshot

**Solution architecture**

Figure 1 shows the architecture of the solution, which is composed of the following layers:

- **Compute layer**—Comprises fourteen servers. Ten of them are installed and configured as a VMware ESXi 5.5 cluster and four are physical servers.

- **Network layer**—Comprises two IP switches and two director-class SAN switches. The SAN switches are designed for deployment in storage networks supporting virtualized data centers and enterprise clouds.

- **Storage layer**—Comprises an XtremIO array that is configured with a single X-Brick and later scaled to four X-Bricks with 60.9 TB of usable physical capacity. The storage layer also includes the XtremIO Storage Management Application, a powerful and visually intuitive XtremIO system dashboard that is used to view the performance, capacity, and system health of the array.

Figure 1. Solution architecture

Two physical servers are used to create the two-node Oracle 12c RAC database with the production workload running on it, and two others are used to run the test/dev workloads. Also, ten physical servers were configured as the VMware ESXi HA cluster on which VMware vSphere virtual machines were created.
Table 1 lists the hardware resources used in this solution.

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Quantity</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage array</td>
<td>1</td>
<td>XtremIO array consisting of four X-Bricks</td>
</tr>
<tr>
<td>Servers</td>
<td>14</td>
<td>16 cores, 2.9 GHz processors, 512 GB RAM, including: 1 x 1 Gb Ethernet (GbE) network interface card (NIC) 2 x 10 GbE NIC</td>
</tr>
<tr>
<td>LAN switches</td>
<td>2</td>
<td>10 GbE</td>
</tr>
<tr>
<td>SAN switches</td>
<td>2</td>
<td>FC</td>
</tr>
</tbody>
</table>

Table 2 lists the software resources used in this solution.

<table>
<thead>
<tr>
<th>Software resources</th>
<th>Version</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMware vSphere</td>
<td>5.5</td>
<td>Hypervisor hosting all virtual machines</td>
</tr>
<tr>
<td>VMware vCenter</td>
<td>5.5</td>
<td>vSphere manager</td>
</tr>
<tr>
<td>Oracle Enterprise Linux</td>
<td>6.5</td>
<td>Operating system for database servers</td>
</tr>
<tr>
<td>Oracle Database 12c Release 1</td>
<td></td>
<td>Database</td>
</tr>
<tr>
<td>Oracle Grid Infrastructure 12c Release 1</td>
<td></td>
<td>Clusterware with ASM for volume management</td>
</tr>
<tr>
<td>Oracle Database 11g Release 2</td>
<td></td>
<td>Database</td>
</tr>
<tr>
<td>Oracle Grid Infrastructure 11g Release 2</td>
<td></td>
<td>Clusterware with ASM for volume management</td>
</tr>
<tr>
<td>Silly Little Oracle Benchmark (SLOB)</td>
<td>2.2</td>
<td>Random I/O benchmark tool</td>
</tr>
<tr>
<td>XIOS</td>
<td>3.0.0-44</td>
<td>XtremIO operating system</td>
</tr>
<tr>
<td>Swingbench</td>
<td>2.5.0.952</td>
<td>OLTP benchmark tool</td>
</tr>
<tr>
<td>PowerPath</td>
<td>5.7 SP 5 (build 2)</td>
<td>Multipathing software used in physical environment</td>
</tr>
<tr>
<td>PowerPath/VE</td>
<td>5.9 SP 1 (build 54)</td>
<td>Multipathing software used in virtualized environment</td>
</tr>
<tr>
<td>AppSync</td>
<td>2.1.0.0</td>
<td>Advanced copy management software for EMC storage arrays</td>
</tr>
</tbody>
</table>
Storage layer: EMC XtremIO 3.0

XtremIO features

This solution includes the following XtremIO features:

- **Inline data reduction**—XtremIO ensures that duplicate data blocks never translate into physical data writes and are replaced with in-memory metadata pointers that allow a single physical block on SSD to be referenced multiple times.

- **Thin provisioning**—XtremIO allocates capacity to volumes on demand in fine-grained increments. This feature automatically matches capacity allocations to host, operating system, and application demands for maximum efficiency without any post-reclamation operations or any impact on performance.

- **Inline data compression**—XtremIO automatically compresses data after all duplications have been removed. This feature ensures that the compression is performed only for unique data blocks. Data compression is performed in real time and not as a post-processing operation.

- **XtremIO snapshots**—XtremIO snapshot technology is implemented by using the content-addressing capabilities of the array and the in-memory and dual-stage metadata of the system. XtremIO snapshots are optimized for SSD media with a unique metadata tree structure that directs I/Os to the right data timestamp.

- **Fault protection**—XtremIO delivers reliability and availability with redundant components and the ability to tolerate any component failure without a loss of service.

- **Scale-out**—With the combination of the X-Brick building block, the RDMA Infiniband fabric combined with in-memory global metadata, and the XtremIO operating system (XIOS) software, XtremIO enables linear increase in both aggregate capacity and aggregate performance with every additional X-Brick in the cluster.

- **In-memory metadata operations**—XtremIO leverages its multi-controller scale-out design and direct memory-to-memory RDMA fabric to maintain all metadata in memory. Heavy metadata operations such as inline deduplication, thin provisioning allocations, and internal array copy operations are conducted entirely in memory, at instantaneous speed, without impacting I/O to the SSD portion of the array.

- **XtremIO data protection (XDP)**—The XtremIO data protection scheme is different from traditional RAID protections. It provides high efficiency, "self-healing", and double-parity data protection. The cluster requires very little capacity overhead for data protection and metadata space.

Storage configuration

For this solution, XtremIO was first deployed in a single X-Brick configuration and was then scaled to a four X-Brick cluster with built-in, redundant, 40 Gb/s QDR InfiniBand switches providing back-end connectivity between the storage controllers. This ensures a highly available, ultra-low latency RDMA network.
With XtremIO data protection, XtremIO requires far less reserved capacity for data protection, metadata storage, snapshots, spare drives, and performance, leaving much more space for user data. This lowers the cost per usable GB. In this 4-X-Brick configuration, the XtremIO cluster was configured with one hundred 800 GB SSDs, which provides 60.9 TB usable physical space.

**Storage design**

Traditional storage designs for an Oracle Database use multiple RAID groups of different drive types that are created with different levels of protection for each drive type, and the protection is distributed across multiple controllers.

With XtremIO, all drives are under XDP protection, which uses both the random access nature of flash and the unique XtremIO dual-stage metadata engine. This means that data blocks in the array are distributed evenly across the X-Bricks to maintain consistent performance, extending the longevity of the flash drives.

Databases generate both random and sequential I/O, as shown in Figure 2. With XtremIO, these are treated equally, because data is randomized and distributed in a balanced fashion throughout the array.

![Figure 2. Database front-end random and sequential I/O](image)

For more information about EMC XtremIO 3.0, refer to the References section of this white paper.
Virtualization layer: ESXi and virtual machine

The choice of a server platform for a virtualized infrastructure is based on both the supportability of the platform and the technical requirements of the environment. In production environments, the servers must have:

- Sufficient cores and memory to support the required number and workload of the virtual machines
- Sufficient connectivity, both Ethernet and FC, to enable redundant connectivity to the IP and storage network switches
- Sufficient capacity to withstand a server failure and support failover of the virtual machines

In this solution, we used ten physical servers configured as a vSphere HA cluster, with each running a vSphere ESXi server. We then deployed virtual machines to create multiple virtualized Oracle databases with different versions/releases, including both 11g and 12c.

For further information about recommended practices for VMware virtualization, refer to the References section.
Oracle Database

Overview

For this solution we created two sets of databases as follows:

- **Physical databases**—Three 12c physical databases were created:
  - One 20 TB two-node Oracle 12c RAC database used as the production database
  - One single-instance Oracle 12c database created from XtremIO snapshots of the physical production database and used for development purposes
  - One single-instance Oracle 12c database created from XtremIO snapshots of the physical production database and used for testing purposes

- **Virtualized databases**—Five single-instance databases were created for different test scenarios:
  - One 1 TB 11gR2 database used as a production database
  - One 1 TB 12cR1 database used as a production database
  - One 1 TB 12cR1 database with a multitenant option enabled used as a production database
  - One 12c database created from XtremIO snapshots of the 12cR1 database and used for development and testing purposes
  - One 12c database used to validate the interoperability of XtremIO compression and Oracle Advanced Compression

---

**Note:** The production databases used for either the physical or virtual environments were configured with archive logging enabled to simulate real-world cases.

*Oracle Best Practices with XtremIO* provides more information about configuring Oracle databases on XtremIO.

---

2 “We” refers to the EMC Engineering Team who tested the solution.
Use case 1: Scale out and consolidate production and non-production environments

Description
In this use case, we ran the SLOB workload on a physical two-node RAC production database built on a single X-Brick XtremIO array. Performance data, including IOPS was recorded as a baseline.

The XtremIO array was then scaled out to a four X-Brick configuration and the workload was quadrupled. Performance results from the four X-Brick configuration were compared to the performance result from the single X-Brick configuration. This test is intended to verify that, as the XtremIO array scales from a single X-Brick to four X-Bricks and the workload is quadrupled, the IOPS is also quadrupled.

On the four-X-Brick XtremIO configuration, to provision the test/dev environment, we created two cross-consistent XtremIO snapshots of the physical production database and mounted them to two physical servers as test/dev environments. The production and the test/dev workloads were then run in parallel to verify that there was minimal performance impact on the production database.

Figure 3 shows the logical architecture for use case 1.

Figure 3. Logical architecture for testing the scalability and workload of XtremIO

Configuration
In this use case, we configured a two-node RAC production database on two physical servers. Two more physical servers were configured as single instance databases and used for test/dev purposes. During testing, the number of X-Bricks in the XtremIO array was scaled from one to four.
The XtremIO volumes and ASM disk groups for both configurations are shown in Table 3.

Table 3.  
XtremIO volumes and ASM disk groups used by the physical production database

<table>
<thead>
<tr>
<th>ASM disk groups</th>
<th>Volume size (GB)</th>
<th>Number of volumes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>+DATA</td>
<td>2048</td>
<td>11</td>
<td>Used for data files, temp files, and control files</td>
</tr>
<tr>
<td>+REDO</td>
<td>40</td>
<td>4</td>
<td>Used for online log files</td>
</tr>
<tr>
<td>+FRA</td>
<td>500</td>
<td>2</td>
<td>Used for archived log files</td>
</tr>
</tbody>
</table>

Table 4 details the physical production database and workload profile for this use case.

Table 4.  
Physical production database configuration and workload profile

<table>
<thead>
<tr>
<th>Profile characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database type</td>
<td>OLTP</td>
</tr>
<tr>
<td>Database size</td>
<td>20 TB</td>
</tr>
<tr>
<td>Oracle Database</td>
<td>Two-node Oracle 12c R1 RAC database on ASM</td>
</tr>
</tbody>
</table>
| Instance configuration | SGA size: 16 GB  
**Note:** Because larger database cache size will buffer more data, we configured a very small buffer cache to generate a stable and high physical I/O workload. |
| Workload profile       | SLOB random I/O workload with 80:20 read/write ratio and SLOB execution think time enabled. Refer to the Appendix in this paper for a full list of SLOB configuration parameters used. |
| Data block size        | 8 KB        |
We used the following steps to perform the scale-out and consolidation test:

1. Configure the single X-Brick XtremIO by creating volumes, mapping them to the XtremIO array initiator group, and so on. Refer to Storage configuration for information about creating and mapping the volumes to the initiator groups.

2. Deploy a two-node physical Oracle RAC 12c database that is populated with 20 TB SLOB data as the production database.

3. Run the SLOB workload with 18 concurrent sessions on the database and record the performance statistics.

4. Scale the single X-Brick XtremIO to four X-Bricks.

5. Run the SLOB workload with 72 sessions on the database and record the performance statistics.

6. Create two XtremIO cross-consistent snapshots for the volumes that belong to the production database.

7. Mount the snapshots separately to two physical servers to set up two single instance databases that will be used as the test/dev environments.

8. Run the same SLOB workload against the production database and simultaneously run the simulated test/dev workloads on the test/dev databases to verify that the test/dev database workload had no performance impact on the production workload.

Note: The single X-Brick XtremIO array that is used in this solution provides 15.2 TB of usable physical space. With XtremIO in-line Data Reduction Services, including thin provisioning, inline global deduplication, and compression, we were able to create a database populated with 20 TB of data on the single X-Brick XtremIO array.

Note: Snapshots on a set of volumes can be created manually by selecting a set of volumes or by placing volumes in a consistency group container folder and creating a snapshot of the consistency group folder.

Results

OLTP database workload performance metrics

For tests that include the SLOB workload, we used the following statistics from the Automatic Workload Repository (AWR) reports to validate the workload performance.

Figure 4 shows performance data from the AWR report created during the single X-Brick XtremIO test.
The following key metrics were extracted from the AWR Report, as shown in Figure 4.

- **Physical write I/O requests** shows 3,181 physical write IOPS and corresponds to “Write IOPS” in Table 5.
- **Physical read I/O requests** shows 12,651 physical read IOPS and corresponds to “Read IOPS” in Table 5.
- **Redo size** is 2,730,371 bytes per second, which is 3MB/s of redo write I/O bandwidth, corresponding to “Redo throughput (MB/s)” in Table 5.

In the test case, we observed the physical read I/O total requests as 12,659 per second, and the physical write I/O total requests as 3,284 per second. There was no RMAN or other extraneous I/O as there might be in real application systems. We disabled Oracle maintenance scheduler jobs, so the number of physical read total I/O requests is was approximately equal to physical read I/O requests, and the number of physical write total I/O requests is approximately equal to physical write I/O requests.

All I/O is important to the DBA but we focused on application I/O activity as a way to narrow the analysis. We used physical read I/O requests and physical write I/O requests in our analysis, as they only include Application IOs.
Test results: Scaling the number of X-Bricks from one to four in the XtremIO array

Table 5 compares the performance statistics of both configurations, and shows that while the database workload was quadrupled, the aggregate IOPS was also quadrupled.

Table 5. IOPS of single X-Brick and four X-Brick XtremIO array

<table>
<thead>
<tr>
<th>Performance metric</th>
<th>Performance data</th>
<th>Single X-Brick Cluster</th>
<th>Four X-Brick Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read IOPS</td>
<td></td>
<td>12,651</td>
<td>47,127</td>
</tr>
<tr>
<td>Write IOPS</td>
<td></td>
<td>3,182</td>
<td>11,683</td>
</tr>
<tr>
<td>Aggregate IOPS (Write + Read)</td>
<td></td>
<td>15,833</td>
<td>58,810</td>
</tr>
</tbody>
</table>

Test results: Consolidating production and TEST/DEV database workloads on the four X-Brick XtremIO array

Two snapshots were created and mounted separately to two physical servers as the TEST and DEV databases.

The following workloads were run to test the performance of the XtremIO array during consolidation:

1. A SLOB workload with 72 concurrent users was run on the production database.
2. A SLOB workload with 10 concurrent users, which simulates the DEV workload, was then run on the DEV database.
3. A SLOB workload with 18 concurrent users, which simulates the TEST workload, was then run on the TEST database.

Table 6 shows the performance impact on the production database as the DEV and TEST workloads were consolidated to the same XtremIO array.

Table 6. Performance statistics of the consolidated production, test, and dev workloads on the same XtremIO array

<table>
<thead>
<tr>
<th>Performance metric</th>
<th>Performance data</th>
<th>PROD only</th>
<th>PROD and DEV</th>
<th>PROD, DEV, and TEST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>PROD</td>
<td>DEV</td>
<td>PROD</td>
</tr>
<tr>
<td>Read IOPS</td>
<td></td>
<td>47,127</td>
<td>6,526</td>
<td>46,987</td>
</tr>
<tr>
<td>Write IOPS</td>
<td></td>
<td>11,683</td>
<td>1,616</td>
<td>11,630</td>
</tr>
<tr>
<td>Aggregate IOPS (Write + Read)</td>
<td></td>
<td>58,810</td>
<td>8,142</td>
<td>58,617</td>
</tr>
</tbody>
</table>

Table 6 shows the performance impact on the existing production database, when the DEV and TEST workloads were consolidated into the XtremIO array.
When the DEV database that was running 8,142 aggregate IOPS was consolidated into the array, the workload and performance of the production database remained almost the same.

When the concurrent workload was run against both the DEV and TEST databases, the impact on the production database performance was minimal, with an aggregate IOPS of 99.6% of the baseline.

**Use case summary**  
In this use case, we demonstrated the following:
- XtremIO can accommodate increasing database workloads by adding more X-Bricks to the array, to provide more IOPS.
- XtremIO snapshots provide a simple way to consolidate production and test/dev workloads onto the same array, while the performance impact of consolidating databases on the production database is kept to a minimum.
Use case 2: Consolidate physical and virtualized databases onto the four-X-Brick XtremIO array

Description

In this use case, three virtual single instance databases were built as virtual production environments. The virtual environments included the following:

- One Oracle 11gR2 database
- One Oracle 12c database
- One Oracle 12c database with the multitenant option enabled.

To validate the performance impact on the physical production database as the virtualized databases were consolidated on the XtremIO array, SLOB workloads were simultaneously run against the three virtualized databases and the physical production database.

We then provisioned a test/dev database in the virtual environment, and validated the performance impact of the following operations on the virtualized production database:

- Deploy a virtual machine from the template, create XtremIO cross-consistent snapshots for the volumes of the virtualized Oracle 12c production database, and mount the snapshots to the virtual machine
- Run a simulated test/dev workload against the provisioned test/dev database

Figure 5 shows the logical architecture of this use case.
Figure 5. Logical architecture for consolidating physical and virtual databases on the same array

Configuration

In this use case, we created two VMware virtual machine templates with 11gR2 and 12c Oracle database software installed.

Table 7 shows the virtual hardware resources that were allocated and the system configurations of the templates.

Table 7. Virtual machine template configuration

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>16 vCPUs</td>
</tr>
<tr>
<td>Memory</td>
<td>40 GB</td>
</tr>
<tr>
<td>Operating system</td>
<td>Oracle Enterprise Linux 6.5</td>
</tr>
<tr>
<td>Kernel</td>
<td>3.8.13-16.2.1.el5uek.x86_64</td>
</tr>
<tr>
<td>Virtual network interfaces</td>
<td>Eth0: Public/management IP network</td>
</tr>
<tr>
<td>Software preinstalled</td>
<td>Oracle Grid Infrastructure (for standalone server)</td>
</tr>
<tr>
<td></td>
<td>Oracle Database</td>
</tr>
</tbody>
</table>
### Part Description

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPM packages installed (as Oracle prerequisites)</td>
<td>As specified in the Oracle installation guide</td>
</tr>
<tr>
<td>Disk configuration</td>
<td>100 GB virtual disk for root, /tmp, swap space, and Oracle Database binaries</td>
</tr>
</tbody>
</table>

For each virtual machine, we created a set of volumes from the XtremIO array, as shown in Table 8:

#### Table 8. ASM disk groups and XtremIO volumes

<table>
<thead>
<tr>
<th>ASM disk group</th>
<th>Volume size (GB)</th>
<th>Number of volumes</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>+DATA</td>
<td>300</td>
<td>4</td>
<td>Used for data files, temp files, and control files</td>
</tr>
<tr>
<td>+REDO</td>
<td>20</td>
<td>2</td>
<td>Used for online log files</td>
</tr>
<tr>
<td>+FRA</td>
<td>500</td>
<td>2</td>
<td>Used for archived log files</td>
</tr>
</tbody>
</table>

With the volumes created for the virtual databases, a 500GB VMFS datastore was also created on an XtremIO volume. This datastore held the virtual machine templates.

**Note:** All volumes created from XtremIO and attached to the ESXi hosts support Hardware Acceleration: VMware vSphere Storage APIs – Array Integration (VAAI). For more information about VAAI, refer to the References section.

Three virtual machines were created from these two templates, each containing a single instance database, as shown in Table 9.

#### Table 9. Virtual machine deployed in this use case

<table>
<thead>
<tr>
<th>Virtual machine</th>
<th>Database name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VM1</td>
<td>slobdb1</td>
<td>Single instance 11g R2 database</td>
</tr>
<tr>
<td>VM2</td>
<td>slobdb2</td>
<td>Single instance 12c database without Multitenant option</td>
</tr>
<tr>
<td>VM3</td>
<td>slobdb3</td>
<td>Single instance 12c database with Multitenant option</td>
</tr>
</tbody>
</table>
Table 10 details the virtual database and workload profile for this use case.

### Table 10. Virtual databases configuration and workload profile

<table>
<thead>
<tr>
<th>Profile characteristic</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database type</td>
<td>OLTP</td>
</tr>
<tr>
<td>Database size</td>
<td>1 TB</td>
</tr>
<tr>
<td>Oracle Database</td>
<td>Single instance 11gR2/12cR1 database on ASM</td>
</tr>
<tr>
<td>Instance configuration</td>
<td>SGA size: 16 GB</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Because a larger database cache will buffer more data, we configured a very small buffer cache to generate a stable and high physical I/O workload.</td>
</tr>
<tr>
<td>Workload profile</td>
<td>SLOB random I/O workload with 80:20 read/write ratio and SLOB execution think time enabled. Refer to the Appendix for a full list of SLOB configuration parameters used during the tests.</td>
</tr>
<tr>
<td>Database block size</td>
<td>8 KB</td>
</tr>
</tbody>
</table>

**Testing detail**

We used the following steps to validate the consolidation of physical and virtual databases on the same XtremIO storage array:

1. Use the same physical two-node Oracle RAC database that was deployed in Use case 1: Scale out and consolidate production and non-production environments.
2. Deploy three virtual machines from templates, as shown in Table 9.
3. Create three Oracle databases on the virtual machines separately and populate 1 TB of data to each of the databases using SLOB.
4. Run the SLOB workload on the physical production database to gather the performance baseline.
5. Run the SLOB workload on the physical production database, simultaneously run the same workload on the virtual production database slobdb1, and gather the performance statistics.
6. Run the SLOB workload on the physical production database, simultaneously run the same workload on two virtual production databases (slobdb1 and slobdb2), and gather the performance statistics.
7. Run the SLOB workload on the physical production database, simultaneously run the same workload on three virtual production databases (slobdb1, slobdb2, and slobdb3), and gather the performance statistics.
8. Run the SLOB workload on database slobdb2, provision a test/dev database by deploying the fourth virtual machine from the template, create an XtremIO cross-consistent snapshot of database slobdb2, and mount the snapshot to the virtual machine.

9. Run the SLOB workload on database slobdb2, simultaneously run a simulated test/dev workload on the provisioned test/dev database, and gather the performance statistics.

Results

This section presents the results of consolidating physical and virtual databases on the same XtremIO array.

Table 11 shows the performance impact on the PROD database as more and more databases were consolidated onto the XtremIO array.

<table>
<thead>
<tr>
<th>Performance metric</th>
<th>PROD only</th>
<th>PROD and slobdb1</th>
<th>PROD, slobdb1, and slobdb2</th>
<th>PROD, slobdb1, slobdb2, and slobdb3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read IOPS</td>
<td>47,127</td>
<td>46,513</td>
<td>45,323</td>
<td>44,602</td>
</tr>
<tr>
<td>Write IOPS</td>
<td>11,683</td>
<td>11,506</td>
<td>10,211</td>
<td>10,027</td>
</tr>
<tr>
<td>Aggregate IOPS (Write + Read)</td>
<td>58,810</td>
<td>58,019</td>
<td>55,085</td>
<td>54,630</td>
</tr>
</tbody>
</table>

When four databases are run in parallel on the four-X-Brick XtremIO array, the following performance results are verified, as shown in Figure 8.

- When slobdb1 with a workload of 55,085 aggregate IOPS was added to the array, the total aggregate IOPS generated increase to 113,104.
- When slobdb2 with a workload of 54,545 aggregate IOPS was further added into the array, the total aggregate IOPS generated increased to 164,454.
- When slobdb3 with a workload of 53,238 aggregate IOPS was further added into the array, the total aggregate IOPS generated increased to 209,269.

Figure 6 is a screenshot taken from the XtremIO monitoring console, which details the 8 KB block read/write I/O response time when the four databases are running in parallel on the XtremIO array.
Figure 6. 8 KB blocks read/write I/O response time when the four databases ran in parallel

As Figure 6 shows, on the array side, when the four databases run in parallel, the 8 KB block read/write I/O response time is less than 1 ms.

Figure 7 is taken from the XtremIO monitoring console, which shows the array CPU utilization while both the physical and virtualized databases are running in parallel.
Figure 7. Array CPU utilization as physical and virtualized databases run in parallel

Figure 7 shows that, as the physical and virtualized databases were consolidated on the four-X-Brick XtremIO array generating 209,269 aggregate IOPS, the CPU utilization of the array was less than 40%.

Note: In our configuration, the two physical servers of the physical production RAC database were connected to all the FC front-end ports of the array, and the ten ESXi servers were connected to a subset of the FC front-end ports of the array. Figure 7 shows CPU utilization differences among the storage array engines.

Test results: Provisioning a test/dev database from the virtualized production databases on the four-X-Brick XtremIO array

Table 12 provides the following performance details extracted from the AWR reports:

- A performance baseline of the virtual production database
- The performance statistics of the virtualized production database when a new virtual machine is deployed from the template and a cross-consistent snapshot is subsequently created for the volumes of the virtualized production database
- Performance statistics of both the virtual production database and the test/dev database when they are run in parallel
Table 12. Performance statistics of provisioning a virtual test/dev and consolidating to XtremIO array

<table>
<thead>
<tr>
<th>Performance metric</th>
<th>Performance data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Virtual PROD DB only</td>
</tr>
<tr>
<td></td>
<td>PROD</td>
</tr>
<tr>
<td>Read IOPS</td>
<td>46,473</td>
</tr>
<tr>
<td>Write IOPS</td>
<td>11,483</td>
</tr>
<tr>
<td>Aggregate IOPS (Write + Read)</td>
<td>57,956</td>
</tr>
</tbody>
</table>

As shown in Table 12, test/dev database provisioning had minimal performance impact on the source virtual production database.

The virtual machine and the template it was deployed from are all stored on the same datastore. Because the datastore was created from the XtremIO array, the virtual machine deployment process was accelerated by the VAAI. With VAAI, the copying operation incurred by virtual machine deployment was offloaded to the array. This means that transferring data between the ESXi hosts and the XtremIO array is not necessary. Also, because of the real-time inline data reduction technology and in-memory metadata feature of the array, VAAI allows near-instantaneous cloning and provisioning of virtual machines while reducing the storage footprint required by the provisioned virtual machine.

When we ran a simulated test/dev workload with IOPS 7,930 aggregate to the test/dev database, the test result shows a total aggregate IOPS of 65,486.

Use case summary

In this use case, we demonstrated the following:

- XtremIO can consolidate various versions of physical and virtualized Oracle databases with acceptable performance degradation.

- The provisioning of virtualized test/dev Oracle databases can be accelerated by XtremIO snapshot, XtremIO VAAI support, and VMware templates, while the storage footprint and performance impact on the source database is reduced to a minimum.
Use case 3: Validating how XtremIO compression works with Oracle Advanced Compression

Description

In this use case, we tested how, under different database table PCTFREE settings, the performance of OLTP workload and physical storage capacity was impacted when we enabled or disabled Oracle Advanced Compression Option (ACO) on the database segments stored on the XtremIO array.

We created four tablespaces on separate ASM diskgroups, loaded the same Swingbench Order-Entry schema data separately into the four tablespaces with their different segment settings that are enabled with or without Oracle ACO compression, and set the values (10 or 0) to the PCTFREE of the tables.

Because XtremIO compression is an always-on feature, we compared the space efficiency before and after Oracle Advanced Compression was enabled for the database segments stored on the XtremIO array. To further validate the interoperability of XtremIO compression and Oracle Advanced Compression, we ran the same Swingbench OLTP workload on the segments with and without Oracle ACO compression.

The compatibility between XtremIO compression and Oracle Advanced Compression can be validated through the space efficiency and performance difference.

Figure 8 shows the logical architecture of this use case.
In this use case, we deployed a virtual machine with the default settings from the template created in use case 2 (see Table 7 on page 23) and then created a single instance Oracle 12c database. We created volumes with different sizes and then created VMFS datastores separately on these volumes.

Table 13 details the names and descriptions of the ASM diskgroups and the number and size of the XtremIO volumes we created in this use case:

**Table 13. ASM diskgroups and XtremIO volumes**

<table>
<thead>
<tr>
<th>ASM diskgroup</th>
<th>Volume size (GB)</th>
<th>Number of volumes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>+SYS</td>
<td>60</td>
<td>2</td>
<td>Used for system, sysaux, undo, temp tablespaces and control files</td>
</tr>
<tr>
<td>+SEED</td>
<td>300</td>
<td>4</td>
<td>Used to store the seed data which will be loaded to other tablespaces</td>
</tr>
<tr>
<td>+DATA1</td>
<td>300</td>
<td>4</td>
<td>Used for tablespace in which tables and indexes were not ACO compressed, and tables were created with PCTFREE=10</td>
</tr>
<tr>
<td>+DATA_ACO1</td>
<td>300</td>
<td>4</td>
<td>Used for tablespace in which tables and indexes were ACO compressed, and tables were created with PCTFREE=10</td>
</tr>
<tr>
<td>ASM diskgroup</td>
<td>Volume size (GB)</td>
<td>Number of volumes</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>+DATA2</td>
<td>300</td>
<td>4</td>
<td>Used for tablespace in which tables and indexes were not ACO compressed, and tables were created with PCTFREE=0</td>
</tr>
<tr>
<td>+DATA_ACO2</td>
<td>300</td>
<td>4</td>
<td>Used for tablespace in which tables and indexes were ACO compressed, and tables were created with PCTFREE=0</td>
</tr>
<tr>
<td>+REDO</td>
<td>20</td>
<td>2</td>
<td>Used for online log files</td>
</tr>
<tr>
<td>+FRA</td>
<td>500</td>
<td>2</td>
<td>Used for archived log files</td>
</tr>
</tbody>
</table>

Table 14 details the database configuration and workload profile for this use case.

### Table 14. Database configuration and workload profile

<table>
<thead>
<tr>
<th>Profile characteristic</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database type</td>
<td>OLTP</td>
</tr>
<tr>
<td>Tablespace size</td>
<td>1 TB for each of tablespaces created</td>
</tr>
<tr>
<td>Oracle Database</td>
<td>Single instance 12cR1 database on ASM</td>
</tr>
<tr>
<td>Instance configuration</td>
<td>SGA size: 16 GB&lt;br&gt;&lt;br&gt;&lt;b&gt;Note:&lt;/b&gt; Considering that a larger database cache size will buffer more data, we configured a very small buffer cache to generate a stable and high physical I/O workload.</td>
</tr>
<tr>
<td>Workload profile</td>
<td>Swingbench OrderEntry TPCC-like benchmark, read/write ratio: 60/40</td>
</tr>
<tr>
<td>Database block size</td>
<td>8 KB</td>
</tr>
</tbody>
</table>

#### Testing detail

We used the following steps to validate how Oracle Advanced Compression Option works with XtremIO compression:

1. Deploy a virtual machine from the template with the default configuration, as shown in Table 7.
2. Create an Oracle 12c single instance database.
3. Create a 1 TB tablespace named SEED on ASM diskgroup +SEED, and load the Swingbench Order Entry schema data into a schema named SEED.
4. Make a note of the Physical Capacity Used metric shown in the XtremIO dashboard. For an explanation of each metric displayed on the XtremIO dashboard, refer to Methodology on page 46.
5. Create a 1 TB tablespace named DATA1 on ASM diskgroup +DATA1, and then load the Swingbench Order Entry data from schema SEED into a schema named SOE1 without Oracle ACO compression. The tables in schema SOE1 are created with PCTFREE set to 10.

Note: When loading data, we used CREATE TABLE AS SELECT (CTAS) to load the data from schema SEED into the tables created in the schema SOE1, and then created indexes on the tables of schema SOE1. We loaded SEED schema data into other tablespaces using the same method for each subsequent step.

6. Make a note of the Physical Capacity Used metric shown in the XtremIO dashboard. To obtain the physical capacity delta, compare the numbers shown in Physical Capacity Used before and after the data was loaded. The physical capacity delta indicates how much physical space in the XtremIO array was allocated for the data loaded in the previous step.

7. Create a 1 TB tablespace named DATA_ACO1 on ASM diskgroup +DATA_ACO1. Then load the Swingbench Order Entry data from schema SEED into a schema named SOE2. Oracle Advanced Row Compression is enabled for the tables, and Oracle Advanced Index Compression enabled for the indexes. The tables in schema SOE2 are created with PCTFREE set to 10.

8. Make a note of the Physical Space Used metric from the XtremIO dashboard.

9. Create a 1 TB tablespace named DATA2 on ASM diskgroup +DATA2. Load the Swingbench Order Entry data from schema SEED into a schema named SOE3 without Oracle ACO compression enabled. The tables in schema SOE3 are created with PCTFREE set to 0.

10. Make a note of the Physical Space Used metric shown in the XtremIO dashboard.

11. Create a 1TB tablespace named DATA_ACO2 on ASM diskgroup +DATA_ACO2. Then load the Swingbench Order Entry data from schema SEED into a schema named SOE4, with Oracle Advanced Row Compression enabled for the tables and Oracle Advanced Index Compression enabled for the indexes. The tables in schema SOE4 are created with PCTFREE set to 0.

12. Make a note of the Physical Space Used metric from the XtremIO dashboard.

13. Run the same Swingbench OLTP workload on the Oracle ACO compression data and the non-ACO compression data, and record the performance data.

**Results**

Relative physical storage space compression efficiency with and without Oracle ACO enabled on the XtremIO array

We tested the physical storage compression and space efficiency between XtremIO only compression and XtremIO compression with Oracle ACO, with the database table PCTFREE set to either 10 or 0. Figure 9 shows the results.
As XtremIO compression is an always-on feature, all the data must be compressed by XtremIO before it is written to the storage media of the array. This means that when data is loaded to the segments with Oracle ACO enabled, the data will first be compressed by Oracle, and then further compressed by XtremIO as it is written to the XtremIO array.

Figure 9 shows the following results:

- When tables were created with PCTFREE=10, and the same amount of data was loaded separately to schemas SOE1 (without Oracle ACO enabled) and SOE2 (with Oracle ACO enabled):
  - XtremIO reported that data in schema SOE2 required 14.3 percent more array physical space, compared to the data in schema SOE1.
  - This indicates that for the same amount of data loaded, XtremIO compression with Oracle ACO enabled required 14.3 percent more physical space in the array than XtremIO compression alone.

- When tables were created with PCTFREE=0, and the same amount of data was loaded separately to schemas SOE3 (without Oracle ACO enabled) and SOE2 (with Oracle ACO enabled):
  - XtremIO reported that data in schema SOE4 required 16.7 percent less physical space in the array, compared to the data in schema SOE3.
  - This indicates that for the same amount of data loaded, XtremIO compression with Oracle ACO enabled required 16.7 percent less physical space in the array than XtremIO compression alone.
The results show that Oracle ACO and XtremIO compression can co-exist. They also show that when tables are created with PCTFREE set to 0, XtremIO is able to achieve more space saving by further compressing the data that has already been compressed by Oracle ACO.

**OLTP database workload performance with and without Oracle ACO enabled on XtremIO**

Oracle Database compresses blocks of data in batch mode rather than compressing data every time a write operation takes place. This means that a newly initialized block remains uncompressed until data in the block reaches an internally controlled threshold. When a transaction causes the data in the block to reach this threshold, all contents of the block are compressed. Subsequently, as more data is added to the block and the threshold is reached again, the entire block is recompressed to achieve the highest level of compression.

For the OLTP database workload, the test results show negligible differences in performance between XtremIO compression with Oracle ACO and XtremIO compression without Oracle ACO.

According to the document *Oracle Advanced Compression with Oracle Database 12c*, Advanced Row Compression has no adverse impact on read operations since it is able to read compressed blocks directly in memory without uncompressing the blocks. While there can be an additional performance overhead for write operations, Oracle has made several optimizations that minimize this performance overhead for write operations with Advanced Row Compression.

**Use case summary**  
In this use case, we demonstrated that:

- Oracle Advanced Compression Option and XtremIO compression are compatible with each other.
- The physical storage savings on the array can be varied by the database table PCTFREE setting.
- The difference in the OLTP workload performance between XtremIO compression on its own and XtremIO compression combined with Oracle ACO is negligible.
Use case 4: Validating the storage capacity and performance overhead of XtremIO snapshots

Description

In this use case, we validated the performance impact of XtremIO snapshots on the database random I/O workload, both during and after snapshot creation. After that, we provisioned 63 DEV databases using AppSync. To validate the performance impact on the production workload, we ran DEV workloads that were simulated by SLOB executing a SELECT/UPDATE ratio of 25:75 to these DEV databases. We also validated the storage capacity overhead of XtremIO snapshots for both physical and virtualized databases.

Note: Because we set a small database buffer cache to these DEV databases with an active data set that was much larger than the database buffer cache, the UPDATE SQLs processing sequence was essentially read/modify/write in nature. Thus the physical read/write IO ratio of the simulated DEV workloads was 4:3.

Figure 10 shows the logical architecture of this use case.

Figure 10. Logical architecture for testing the storage capacity and performance of XtremIO snapshot
In this use case, we used the databases that were created in the previous tests, including the 20 TB physical Oracle 12c two-node RAC production database, the 1 TB virtual Oracle 11g single instance production database, and the 1 TB virtual Oracle 12g single instance production database. We also used the same SLOB workload from the previous tests. In order to deploy virtual machines for provisioning DEV databases, we used an existing virtual machine template. Refer to the configuration section of Use case 1: Scale out and consolidate production and non-production environments and Use case 2: Consolidate physical and virtualized databases onto the four-X-Brick XtremIO array for more information.

Creating snapshots from the XtremIO Storage Management Application console

Creating an XtremIO snapshot is intuitive and simple; using the XtremIO Storage Management Application console, the snapshots can be created with a few clicks.

You can take a cross-consistent snapshot of the 20 TB physical production database as follows:

1. Select Configuration on the XtremIO Management Application console, right-click the folder that contains all of the volumes of the 20 TB physical production database, and select Create Snapshots, as shown in Figure 11.

2. Specify the Snapshot suffix, select a folder for the snapshot and then click Create, as shown in Figure 12.
Figure 12. Create XtremIO snapshot – step 2

3. When the snapshot creation is complete, the new snapshot appears in the folder you specified, as shown in Figure 13.
Creating snapshots from the AppSync console

Creating snapshots from the AppSync console is intuitive and simple. For XtremIO storage arrays, AppSync uses the XtremIO snapshot feature to create copies of the database to protect and/or repurpose the database.

Before creating snapshots, you need to register the servers that host the database to AppSync and discover the database currently running on these servers. If the server hosting the database is a virtual machine, you need to register the corresponding vCenter server to AppSync before you add the server. After that, you can create copies of the database either for protection or repurposing. We used the following steps to create the repurpose copies for the database:

1. Select **Copy Management** on the AppSync console and click **Oracle**, as shown in Figure 14.

   ![Figure 14. Creating repurpose copy of database using AppSync – step 1](image-url)

2. Select the database for which you want to create a repurpose copy, click **Repurpose**, and select **Create Repurpose Copy** in the pop-up list, as shown in Figure 15.
3. To make the repurposing copies refreshable, click **Create 1st gen copy and a 2nd gen copy**. You can provision a TEST/DEV environment using the second copy and refresh the TEST/DEV environment using the first gen copy later. Click **Next**, as shown in Figure 16.

4. For **Select the Copy Type**, click **Next** to accept the default option, **Snap**, as shown in Figure 17.
5. Select Run Now to create the repurpose copy immediately and click Next, as shown in Figure 18.

6. Review the information shown in the summary table and click Finish to initiate the creation of the repurpose copy, as shown in Figure 19.
7. When complete, the information shown in Figure 20 indicates that the first and second repurpose copies were successfully created.

![Figure 20. Repurpose copies successfully created](image)

**Mount the repurpose copy to the target server to provision TEST/DEV database**

Before mounting the repurpose copy, you need to register the target server to AppSync. If the target server is a virtual machine, you need to register the corresponding vCenter server to AppSync before you register it. After you register the target server to AppSync, you can mount the repurpose copy to the target server on the AppSync console. Use the following procedure to mount the repurpose copy:

1. Click the source database, as shown in Figure 21.

![Figure 21. Mount repurpose copy of database using AppSync – step 1](image)

2. In the list of repurpose copies displayed in the console, select the one you want to mount and click **Mount**, as shown in Figure 22.
Figure 22. Mount repurpose copy of database using AppSync – step 2

3. Verify the information for the repurpose copy and click **Next**, as shown in Figure 23.

Figure 23. Mount repurpose copy of database using AppSync – step 3

4. In the **Mount Settings** box, select the server where you plan to mount the repurpose copy and click **Next**, as shown in Figure 24.
Figure 24.  Mount repurpose copy of database using AppSync – step 4

5. Verify the summary information and click **Finish**, as shown in Figure 25.

Figure 25.  Mount repurpose copy of database using AppSync – step 5

The following information is shown in the boxes, indicating that the repurpose copy is successfully mounted to the target server, as shown in Figure 26.
We used the following steps to validate the performance impact of the XtremIO snapshot on the database random I/O workload (steps 1 – 9) and the storage capacity overhead (steps 10 – 16).

1. Run a SLOB workload on the 20 TB physical production database to get a performance baseline.

2. Run the same workload on the physical production database, and then create a first gen copy and a second gen copy of the database using AppSync as the workload is running.

3. Collect the performance statistics from the Oracle AWR reports. To validate how the workload is affected during snapshot creation, take screenshots from the XtremIO Storage Management Application Monitoring console.

4. Create an additional 62 second gen copies for the physical production database using AppSync.

5. Run the same workload against the physical production database that has a total of 63 second gen copies and one first gen repurpose copy (a copy is a cross-consistent snapshot of the database, which means 64 cross-consistent snapshots are created on its volumes), and record the performance statistics.

6. Deploy 63 virtual machines and mount the 63 second gen repurpose copies to the virtual machines separately to provision 63 DEV databases.

7. Run simulated DEV workloads with a SELECT/UPDATE ratio of 25:75 to these 63 DEV databases, and simultaneously run the same workload against the physical production database.

8. Collect the performance statistics from the Oracle AWR reports.

9. Delete all of the repurpose copies that have been created on the XtremIO array.

10. Create a screen capture of the current space usage of the XtremIO array as the capacity baseline.
11. Create 64 cross-consistent snapshots for the 1TB virtual 12c production database.
12. Create the second screenshot of the current space usage of the XtremIO array.
13. Create 64 cross-consistent snapshots for the 1TB virtual 11g production database.
14. Create the third screenshot of the current space usage of the XtremIO array.
15. Create 64 cross-consistent snapshots for the 20TB physical 12c production database.
16. Create the fourth screenshot of the current space usage of the XtremIO array.

**Note:** Repurposing creates a multi-level tree of copies of the database. Repurpose copies are identified by a generation removed from the production source data, and thus named Gen 1, Gen 2, and so on. A first generation copy creates a copy that can be used as source for other copies (copy-of-copy). Repurpose copies are meant to be mounted for extended periods of time for your various purposes. After use, they can be either discarded or refreshed by synchronizing with the first gen copy.

### Methodology

To validate the space efficiency of the XtremIO array, we used statistics from the dashboard of the XtremIO Storage Management Application, as shown in Figure 27.

![Figure 27. XtremIO dashboard–storage](image)

**Results**
The results are as follows:

- **Volume Capacity** (box A) shows the total amount of space taken by all of the volumes created in the array.

  Note: Volumes in XtremIO array are all thin provisioned.

- **Physical Capacity Used** (box B) shows the amount of physical space allocated in the array; this is the amount of space taken after compression and deduplication.

- **Volume Capacity used** (box C) shows the amount of space allocated in the volumes; this is the amount of space taken before compression and deduplication.

**Test results: Performance impact of XtremIO snapshots on the production database**

We validated the performance impact of snapshots on the production database in the following way.

We first achieved a performance baseline by running SLOB workload with an 80:20 read/write ratio on the production database without a snapshot. We then ran the same workload on the production database and created a first gen and second gen repurpose copy for the databases using AppSync. After 64 cross-consistent snapshots (including one first gen and 63 second gen repurpose copies) were created for the database, the same workload was run on the database and the performance data was recorded. Table 15 shows the performance statistics of the production database we recorded when the workload was running in the following scenarios: without a snapshot, during snapshot creation, and after 64 snapshots were created.

**Table 15. Performance impact of the XtremIO snapshots on the production database**

<table>
<thead>
<tr>
<th>Performance metric</th>
<th>Performance data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Baseline (no snapshot)</td>
</tr>
<tr>
<td>Read IOPS</td>
<td>47,180</td>
</tr>
<tr>
<td>Write IOPS</td>
<td>11,676</td>
</tr>
<tr>
<td>Aggregate IOPS (Write + Read)</td>
<td>58,856</td>
</tr>
<tr>
<td></td>
<td>Creating snapshots</td>
</tr>
<tr>
<td>Read IOPS</td>
<td>47,344</td>
</tr>
<tr>
<td>Write IOPS</td>
<td>11,734</td>
</tr>
<tr>
<td>Aggregate IOPS (Write + Read)</td>
<td>59,078</td>
</tr>
<tr>
<td></td>
<td>After 64 snapshots were created</td>
</tr>
<tr>
<td>Read IOPS</td>
<td>47,181</td>
</tr>
<tr>
<td>Write IOPS</td>
<td>11,684</td>
</tr>
<tr>
<td>Aggregate IOPS (Write + Read)</td>
<td>58,865</td>
</tr>
</tbody>
</table>

As shown in Table 15, XtremIO snapshots imposed little or no performance impact on the production database, either during snapshot creation or after the snapshots were created. For example:

- Before taking the snapshot of the database, the SLOB workload generated 58,856 aggregate IOPS.
- When creating a snapshot with the same workload running, 59,078 aggregate IOPS were generated.
- When 64 snapshots were created, the same workload generated 58,865 aggregated IOPS with an average read response time of 0.481ms.

Figure 28, Figure 29, Figure 30, and Figure 31 show IOPS and I/O response times captured on the XtremIO Monitor console during creation of an XtremIO cross-consistent snapshot with the SLOB workload running.

As Figure 28 shows, I/Os generated by the SLOB workload were largely 8KB in size. The snapshot creation operation had no impact on the aggregate read and write IOPS of the production database.

**Figure 28.  IOPS by block size during snapshot**

Figure 29 provides a closer view of the performance impact of the snapshot creation on the 8KB read/write IOPS. It confirms that snapshot creation had no impact on the read/write IOPS of the SLOB workload.
Figure 29. 8KB block read and write IOPS during snapshot creation

Figure 30 shows that snapshot creation did not impact the read/write I/O response time of the SLOB workload.

Figure 30. 8KB block read/write I/O response time during snapshot creation
Figure 31 shows that snapshot creation did not impact the write I/O response time of each volume used for the online redo log.

Figure 31. Redo write I/O response time during snapshot creation

**Note:** We created the Oracle AWR snapshot before the workload was started, and created the Oracle AWR snapshot after the workload stopped. The I/O response time spike shown in Figure 30 and Figure 31 is caused by Oracle AWR snapshot creation.

XtremIO snapshots are inherently writable and have the same data services and performance as any other volumes in the cluster. When creating snapshots, XtremIO only needs to generate a pointer to the ancestor metadata of the actual volumes in the array. This is a memory-only operation; SSD access and metadata copy operations are not required. As a result, creating an XtremIO snapshot is almost instantaneous and has no performance impact on the source volumes.
Test results: Performance impact of the DEV workloads on the production workload

We validated the performance impact on the production workload by running DEV workloads that were simulated by SLOB sessions. The SLOB sessions executed a set of SQL statements with a mix of 25% SELECT and 75% UPDATE to the databases. We also increased the DEV workloads by increasing the number of SLOB sessions running on each DEV database and validated the performance impact on the production workload.

Table 16 shows the IOPS of the production database as well as the total number of IOPS of all the DEV databases generated with different numbers of SLOB sessions running on them.

Table 16. Performance impact of the DEV workloads on the production workload

<table>
<thead>
<tr>
<th>Performance metric</th>
<th>Performance data</th>
<th>Baseline</th>
<th>PROD and 63 DEV DBs (with 1 SLOB session on each DEV DB)</th>
<th>PROD and 63 DEV DBs (with 2 SLOB session on each DEV DB)</th>
<th>PROD and 63 DEV DBs (with 4 SLOB session on each DEV DB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read IOPS</td>
<td>PROD</td>
<td>47,836</td>
<td>47,185</td>
<td>46,559</td>
<td>70,290</td>
</tr>
<tr>
<td>Write IOPS</td>
<td>PROD</td>
<td>11,896</td>
<td>11,761</td>
<td>11,490</td>
<td>34,960</td>
</tr>
<tr>
<td>Aggregate IOPS</td>
<td>PROD</td>
<td>59,732</td>
<td>58,946</td>
<td>53,508</td>
<td>58,049</td>
</tr>
</tbody>
</table>

Table 16 shows that the workloads running on the 63 DEV databases have little performance impact on the production workload.

- When the simulated DEV workloads with 1 SLOB session ran on each DEV database, the total aggregate IOPS that were generated increased to 112,454.
- When the simulated DEV workloads with 2 SLOB session ran on each DEV database, the total aggregate IOPS that were generated increased to 163,299.
- When the simulated DEV workloads with 1 SLOB session ran on each DEV database, the total aggregate IOPS that were generated increased to 252,704.
Test results: Creating cross-consistent snapshots for the volumes of the databases

Table 17 shows the space efficiency of the XtremIO array as we created cross-consistent snapshots for each database.

### Table 17. Space usage statistics of the array as snapshots were created for the production database

<table>
<thead>
<tr>
<th>Number of snapshots created</th>
<th>Space Statistics</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Volume capacity</td>
<td>Volume capacity used</td>
<td>Physical capacity used</td>
</tr>
<tr>
<td></td>
<td>(TB)</td>
<td>(TB)</td>
<td>(TB)</td>
</tr>
<tr>
<td>Before creating a snapshot</td>
<td>63.4</td>
<td>50.5</td>
<td>13</td>
</tr>
<tr>
<td>64 cross-consistent snapshots for the 1TB virtual 12c Oracle production database</td>
<td>140</td>
<td>50.5</td>
<td>13</td>
</tr>
<tr>
<td>64 cross-consistent snapshots for the 1TB virtual 11g Oracle production database</td>
<td>207</td>
<td>50.5</td>
<td>13</td>
</tr>
<tr>
<td>64 cross-consistent snapshots for the 20TB physical 12c Oracle production database</td>
<td>1690</td>
<td>50.5</td>
<td>13</td>
</tr>
</tbody>
</table>

Table 17 shows that when snapshots are created, the volume capacity of the database increases by the size of the database multiplied by the number of snapshots. However, the amount of used volume capacity and used physical capacity remains the same.

When creating snapshots, the system must only generate a pointer to the ancestor metadata of the actual volume in the array; copying data or metadata is not required, and, as a result, there is no capacity impact on the array. In this use case, 60 TB of usable physical space in the XtremIO array is able to serve 1.65 PB of Oracle databases. When more snapshots are created, this usable physical space is able to serve even more Oracle databases.
Figure 32 shows the space usage before any snapshots are created on the array.

![XtremIO array space usage before creating snapshot](image)

Figure 32. XtremIO array space usage before creating snapshot

Figure 33 shows the space usage after each of the three databases was created with 64 cross-consistent snapshots on the XtremIO array.

![XtremIO space usage after snapshots were created for multiple databases](image)

Figure 33. XtremIO space usage after snapshots were created for multiple databases
Use case summary  In this use case, we demonstrated the following:

- XtremIO snapshots have no performance impact on the database random I/O workload either during or after snapshot creation.
- With AppSync, XtremIO snapshots provide a simple and easy way to provision TEST/DEV and non-production environments.
- The simulated DEV workloads running on the provisioned DEV databases have little to no performance impact on their source database.
- XtremIO snapshots are storage-efficient. As we have shown in this use case, the XtremIO array can support 1.65 PB of Oracle databases with 60 TB of usable physical space.
- The XtremIO array is able to support even more Oracle databases as more snapshots are created.
Conclusion

Summary
This white paper describes a solution that uses EMC XtremIO as the back-end storage for Oracle database servers, which can consist of both physical environments and virtualized environments (using VMware vSphere). The following use cases were explored:

- Scale a single X-Brick XtremIO cluster to a four X-Brick XtremIO cluster while simultaneously quadrupling the production workload.

- The effect of multiple workloads running on the XtremIO array was validated by deploying multiple virtualized Oracle databases on the VMware vSphere environment and running SLOB workloads against these virtualized databases while the physical production workload was running.

- The interoperability of Oracle ACO compression and XtremIO compression was validated by loading the same amount of data into database segments that were created with Oracle ACO compression enabled or disabled and with database table PCTFREE set to different values (0 or 10). The interoperability of the two compression features was further validated by running the same OLTP workload to the segments created.

- The XtremIO snapshot performance overhead was validated by taking multiple cross-consistent snapshots of a production database while running a SLOB workload against that database. The array capacity overhead of XtremIO snapshots was also validated by taking multiple cross-consistent snapshots for Oracle databases with different size, number of volumes, and versions.

Findings
The key findings of the solution tests we conducted are:

- Scaling the back-end XtremIO array from 1 X-Brick to 4-X-Bricks provided perfect linear scaling of IOPS with no statistically significant increase in I/O response time when running an Oracle OLTP workload.

- Creating an XtremIO snapshot of a running Oracle database volume has no performance impact on the workload running against that database.

- The capacity overhead of an XtremIO snapshot at the moment of creation is nil. A very large number of snapshots can thus be supported using XtremIO. This paper proves that 1.65 PB of snapshot copies can be created using only 60 TB of physical storage.

- Consolidating the test/dev workload to the same XtremIO had a negligible performance impact on both the physical and virtual Oracle production database workloads.

- Running multiple virtualized Oracle production databases with OLTP workloads against the XtremIO cluster had a negligible performance impact on the physical production Oracle OLTP workload.
• The combination of Oracle Advanced Compression and XtremIO compression was completely workable, had no significant performance impact, and saved additional physical space.

**Notes:**

• Benchmark results are highly dependent on workload, specific application requirements, and system design and implementation. Relative system performance will vary as a result of these and other factors. Therefore, the solution test workloads should not be used as a substitute for a specific customer application benchmark when critical capacity planning or product evaluation decisions are considered.

• 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 a similar performance expressed in transactions per minute.
References

The following documents, available from the EMC Online Support or EMC.com websites, provide additional and relevant information. If you do not have access to a document, contact your EMC representative:

- **EMC XtremIO High-Performance Consolidation Solution For Oracle**
- **Oracle Best Practices with XtremIO**
- **EMC XtremIO Optimized Flash Storage for Oracle Databases**
- **Introduction to XtremIO Snapshots**
- **XtremIO data protection (XDP)**
- **Introduction to the EMC XtremIO Storage Array (ver. 3.0)**
- **EMC AppSync User and Administration Guide (Version 2.1)**

For additional discussion and information about XtremIO and Oracle, see the following documents:

- **XtremIO Performance Engineering Lab Report: Oracle Database Redo Logging Durability**
- **Lab Report: Oracle Database on EMC XtremIO A Compression Technology Case Study**
- **XtremIO Snapshots – Do you use film or digital photography?**
- **XtremIO In-Memory Metadata**
- **XtremIO Scale-out Design**
- **XtremIO Thin Provisioning**
- **XtremIO Inline Data Reduction**
- **XtremIO Data Protection (XDP)**
- **XtremIO Best Practices: Advanced Format 512e and Native modes**
- **SLOB Deployment – A Picture Tutorial**
- **SLOB Data Loading Case Studies – Part I. A Simple Concurrent + Parallel Example**
- **SLOB Data Loading Case Studies – Part II. SLOB 2.2 For High-Bandwidth Data Loading**
For additional information, see the following documents available from: https://support.oracle.com (sign-in is required)

- Oracle Grid Infrastructure Installation Guide 11g Release 2 (11.2) for Linux
- Oracle Database Installation Guide 11g Release 2 (11.2) for Linux.
- HugePages on Linux: What It Is... and What It Is Not... [My Oracle Support Doc ID 361323.1]
- Oracle Advanced Compression with Oracle Database 12c
- Oracle Database Concepts 12c Release 1 (12.1)
- Grid Infrastructure Installation Guide
- Database Installation Guide

Refer to the following topics on the VMware website:

- Understanding Oracle Certification Support and Licensing in VMware - Environments
- Oracle Databases on VMware Best Practices Guide
- Performance Best Practices for VMware vSphere™ 5.5
- VMware vSphere Storage APIs – Array Integration (VAAI)

Note: The links provided in this document were working correctly at the time of publication.
Table 18 shows the SLOB configuration parameters used in this solution.

Table 18. SLOB configuration parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPDATE_PCT</td>
<td>25</td>
</tr>
<tr>
<td>RUN_TIME</td>
<td>320</td>
</tr>
<tr>
<td>SCALE</td>
<td>400000</td>
</tr>
<tr>
<td>WORK_UNIT</td>
<td>32</td>
</tr>
<tr>
<td>REDO_STRESS</td>
<td>LIGHT</td>
</tr>
<tr>
<td>LOAD_PARALLEL_DEGREE</td>
<td>8</td>
</tr>
<tr>
<td>SHARED_DATA_MODULUS</td>
<td>0</td>
</tr>
<tr>
<td>DO_UPDATE_HOTSPOT</td>
<td>FALSE</td>
</tr>
<tr>
<td>HOTSPOT_PCT</td>
<td>10</td>
</tr>
<tr>
<td>THINK_TM_MODULUS</td>
<td>7</td>
</tr>
<tr>
<td>THINK_TM_MIN</td>
<td>.1</td>
</tr>
<tr>
<td>THINK_TM_MAX</td>
<td>.5</td>
</tr>
</tbody>
</table>