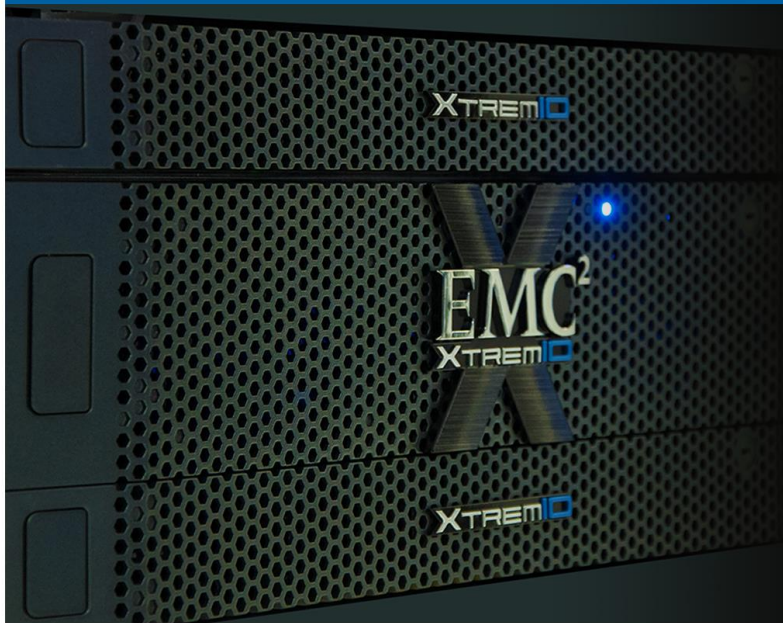


SCALING ORACLE 12c DATABASE PERFORMANCE WITH EMC XTREMIO STORAGE IN A DATABASE AS A SERVICE ENVIRONMENT

Scale out workloads while maintaining database performance with the all-flash EMC® XtremIO™ storage array



Scaled to eight vSphere® VMs running Oracle® Database 12c workloads with minimal impact on IOPS and latencies

Up to 51:1 overall storage efficiency through XtremIO inline deduplication, compression, and thin provisioning

Modern database applications and their supporting infrastructures often have heavy read and write performance requirements. To provide performance and scalability for their users, large organizations, including those providing database-as-a-service (DBaaS) options, now often run database virtual machines (VMs) on flash-based storage arrays.¹ In addition to meeting and maintaining high performance requirements, these storage arrays need to scale the number of VMs and the performance of those VMs while increasing space efficiencies for current and future initiatives. A recent survey indicates that instead of spending time accomplishing these strategic scaling and efficiency objectives, data managers are mired in low-level data administration tasks such as patches, upgrades, and maintaining uptime, and the constant quest for better performance.² The all-flash EMC XtremIO storage array can offer new levels of performance and efficiency for your datacenter and your users, and provides integration tools such as EMC Virtual Storage Integrator for a VMware® vSphere environment to help ease the management burden.

¹ See the report from IDC for EMC: www.emc.com/collateral/analyst-reports/worldwide-all-flash-hybrid-flash-array-forecast-vendor-shares.pdf.

² https://community.emc.com/servlet/JiveServlet/download/37463-15-92338/IOUG_Innovation_FINAL.pdf



In our hands-on testing at Principled Technologies, we measured latency and input/output operations per second (IOPS) supported by the EMC XtremIO array as we scaled from one vSphere VM running a single instance Oracle 12c database to eight identical VMs, all running in a vSphere cluster on four Cisco UCS B200 M3 servers. We found that IOPS levels as reported by the VMs remained consistent as we added database VMs, showing nearly no degradation while the volume of database workloads increased. The consistency demonstrated by this metric suggests the kind of reliable experience users can expect when accessing Oracle 12c databases in an infrastructure supported by an XtremIO array. In addition, we tested the in-line deduplication of the XtremIO storage array to measure capacity savings when cloning a single Oracle 12c VM multiple times. With reduced storage footprints per database VM, the XtremIO array continued to deliver even performance while allowing additional free capacity for additional VM deployments.

Based on our testing, the deduplication capabilities of the XtremIO array can offer support to organizations utilizing a DBaaS model for an increased number of VMs per array while continuing to provide reliable IOPS performance to their users.

HOLD ON TO DATABASE PERFORMANCE WHILE SCALING UP VMS IN YOUR DATACENTER

As your organization grows with more application users, demands placed on your DBaaS infrastructure begin to increase and as a result, IT managers must work with CTOs and CIOs to balance cost, management, and performance. This balance can be difficult to achieve, especially when supporting many virtualized Oracle database environments for production, development, and test workloads. Whether your IT staff needs to control DBaaS offerings for new projects or expand existing Oracle 12c databases due to an increased workload, the storage array your business chooses can directly affect productivity through efficiency and performance. As the performance benefits of flash storage in the datacenter have been widely realized, EMC designed the all-flash-based XtremIO storage array to deliver a platform with simplicity and to reduce tuning for Oracle workloads. Every database gets all-flash storage, which delivered predictable performance and low latency in our tests. We used the Silly Little Oracle Benchmark (SLOB) 2.2 to see how well the all-flash XtremIO storage array could sustain database performance as we increased the number of available database VMs. SLOB is a tool that tests the I/O infrastructure with Oracle SGA-buffered I/O through a set of PL/SQL scripts, and was designed to generate significant I/O impact without stressing the CPU. To learn more about SLOB, see [Appendix A](#).

Designed with multi-controller architecture, the XtremIO array features a dual-stage metadata engine. The array offers “always on” in-line compression and deduplication, and a data protection scheme called XtremIO Data Protection (XDP). In

addition, the array is linearly scalable with clustering, using each additional unit (aka X-Brick) in lockstep. In our testing, we used a single X-Brick. For more information on the EMC XtremIO storage array, see [Appendix A](#). For detailed specification of our test systems, see [Appendix B](#).

TESTING ENVIRONMENT AND PARAMETERS

We cloned eight VMs on our vSphere cluster to run Oracle single instance database workloads. Our goal was to create read and write levels simulating heavy enterprise-level production in each VM to verify the XtremIO array could sustain necessary database performance while linearly scaling at the cluster level as we added VMs. For more information about our database configuration and SLOB, see [Appendix A](#). For detailed testing steps, see [Appendix C](#). For benchmark parameters, see [Appendix D](#).

WHAT WE FOUND

IOPS

Using SLOB to simulate an OLTP-like database, we were able to maintain an average of 14,700 IOPS in each VM as we increased the number of VMs on the servers. As Figure 1 shows, the XtremIO array delivered 118,067 IOPS from all eight VMs without seeing a major degradation in performance on any VM. This output of 118,067 total IOPS is close to EMC's stated theoretical maximum output of the XtremIO storage array in the configuration we tested. Throughout the test duration, the XtremIO storage array averaged 1.5ms write latency and 1.1ms read latency.

Test kits such as SLOB offer an intense I/O load with a minimal SQL-processing CPU cost. In our case, the hosts' CPU average cost was 17 percent and produced 118,067 IOPS on the array. This left 83 percent of the cluster's CPU resources, which would be available for an environment's potentially compute-intensive SQL application code. On the other hand, if the test tool, or a database application, were to consume greater amounts of CPU, then this would leave less CPU capacity for other compute-intensive application code that an environment may require.

Figure 1: In the IOPS tests, the EMC XtremIO array solution delivered consistent performance with each added database VM.

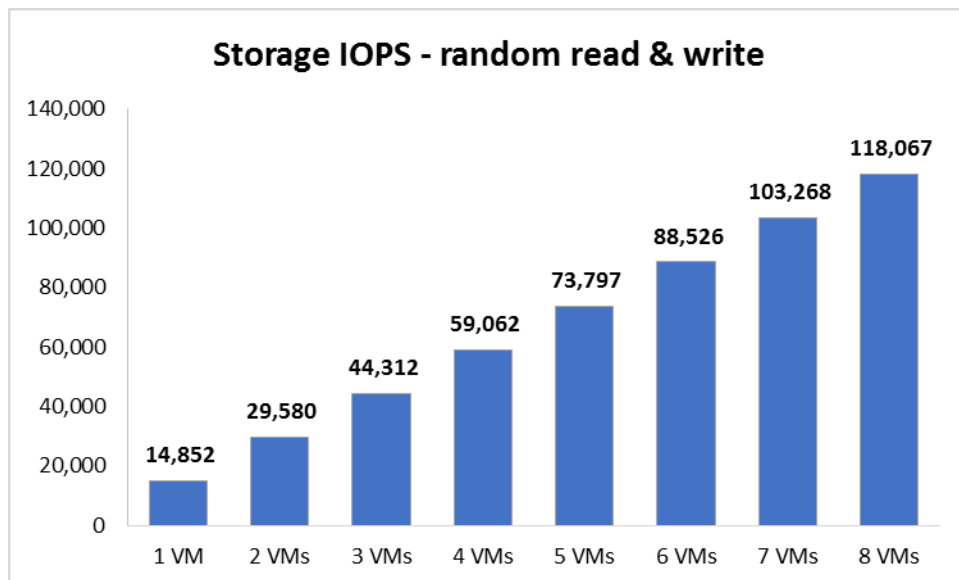


Figure 2 shows the consistency of the IOPS recorded for each database VM during the duration of a single test run. IOPS are reported by iostat from each of the VMs. As each VM started every 10 minutes, the IOPS in each VM stayed consistently around 15,000 IOPS.

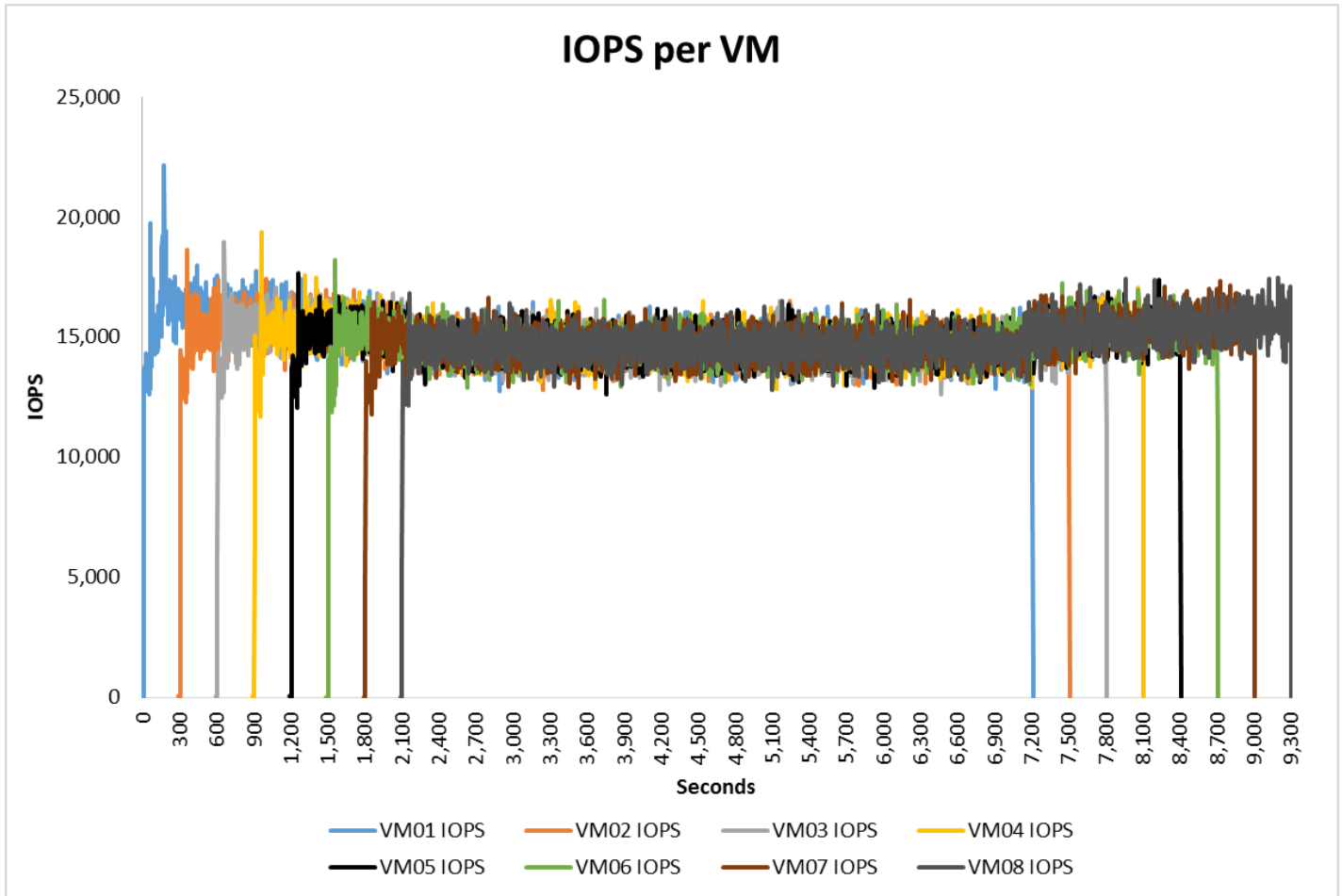


Figure 2: IOPS consistency for each database VM on the EMC XtremIO array solution throughout testing.

Inline data reduction, thin provisioning, and VSI

The XtremIO array natively uses two methods for increasing storage space efficiency: thin provisioning and inline data reduction. Thin provisioning is always used when creating volumes, meaning the system only uses space when it actually needs it and does not reserve empty blocks. In addition, the XtremIO array is content-aware and can place blocks anywhere in the system while still providing great performance with random-access algorithms. This sets the XtremIO thin provisioning apart from traditional disk-based architecture.

The inline data-reduction feature is comprised of two methods: data deduplication and inline data compression. Deduplication refers to the ability of the XtremIO array to remove redundant data as it enters the storage automatically, or “inline,” before writing it to the flash drives. XtremIO offers inline and always-on deduplication, which allows the data deduplication to be performed in the application I/O path. Inline data compression reduces the amount of physical data that needs to be written to disk after the data is deduplicated. This feature further reduces the space

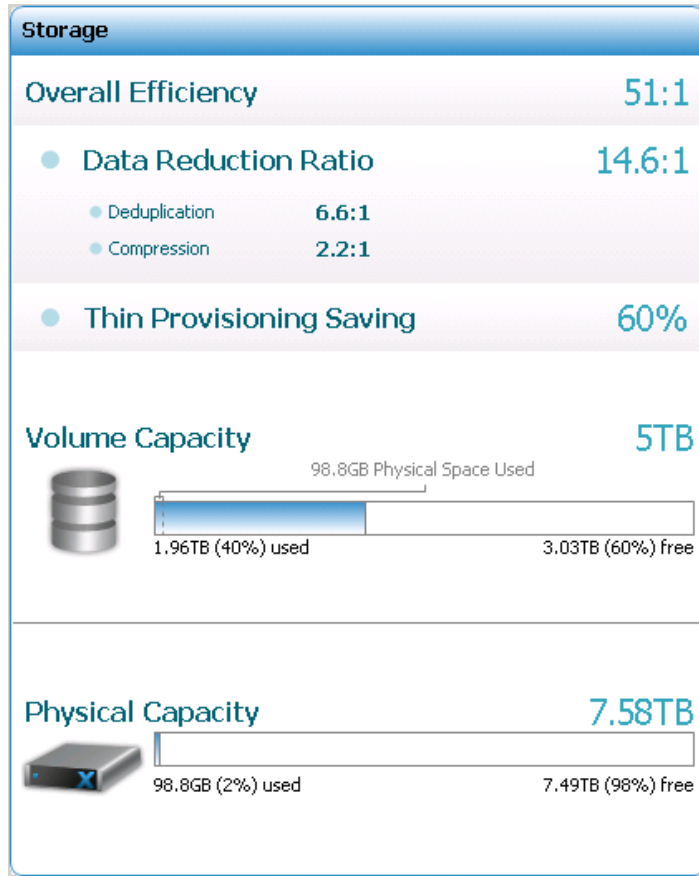
used on the storage and reduces the amount of writes hitting the SSDs. This means that more data can be stored on the array while also increasing the lifespan of the SSDs.

XtremIO reports the deduplication, compression, thin provisioning, and overall efficiency of the entire storage array. In addition, XtremIO provides a Data Reduction Ratio that combines the deduplication and compression ratios and calculates the overall efficiency by dividing the provisioned volume capacity by the actual physical capacity used. We reformatted the XtremIO array to reset the deduplication statistics, created one large LUN, and cloned out eight VMs identical to those that we used in our previous tests. After the cloning, we saw a data reduction ratio of 14.6 to 1 and an overall efficiency of 51 to 1, as seen in Figure 3.

EMC designed the deduplication engine to remove latency overhead, performance impacts, and post-processing operations, meaning that for an Oracle DBA all these benefits could come alongside space-efficient database copy capabilities. Inline deduplication is intended to save DBAs space without performance penalties, and to reduce resource-consuming background processes that could hurt performance.

We used the EMC Virtual Storage Integrator (VSI) plug-in for vSphere in our environment. With the VSI plug-in, you can optimize your hosts, provision storage, clone VMs, and other tasks when using EMC storage arrays such as XtremIO. We used the VSI plug-in to optimize our Cisco blade settings, and to perform the cloning in the deduplication phase of testing.

Figure 3: Storage efficiency statistics from the XtremIO array console after our VM provisioning scenario. XtremIO calculates the overall efficiency by dividing the provisioned volume capacity by the actual physical capacity used.



THE PRODUCTS WE TESTED

About the EMC XtremIO 100% Flash Scale-Out Enterprise storage array

XtremIO is an all-flash scale-out enterprise storage array designed to substantially boost I/O performance. According to EMC, “XtremIO delivers new levels of real-world performance, administrative ease, and advanced data services for applications.” Figure 4 shows the XtremIO user interface.

In this study, one 10TB X-Brick supported the performance of multiple virtualized Oracle databases. We maintained consistent performance and user response times that met our target while increasing the VM load on the storage.

According to EMC specifications, the XtremIO 5TB, 10TB, and 20TB X-Bricks are populated by either 400GB or 800GB eMLC solid-state drives (SSDs). The Starter X-Brick has 13 400GB SSDs for a capacity of 5.2 TB and can be dynamically expanded online up to 10 TB. The 10TB and 20TB X-Bricks have 25 SSDs per system with capacities of 400 GB and 800 GB, respectively. With the current qualified six X-Brick clusters orderable today, that allows for up to 120 TB of physical capacity.³

³ xtremio.com/capacity

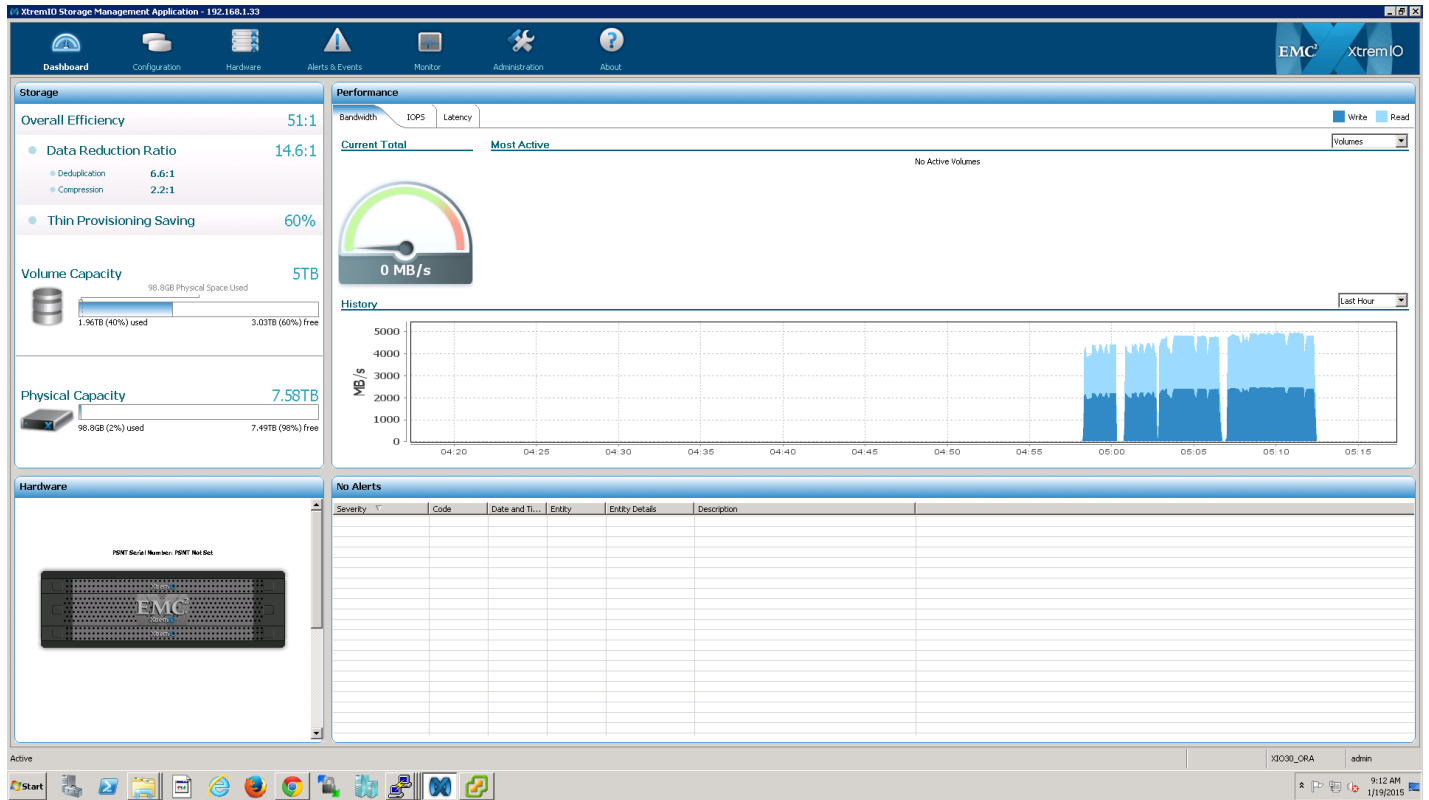


Figure 4: XtremIO Storage Management Application.

As Figure 5 shows, the provisioning and LUN mapping process on XtremIO involves just a few clicks.

For features, XtremIO includes inline data deduplication,⁴ compression, and snapshots as part of its standard offering. Inline data deduplication and compression can potentially translate into significant space savings across databases and applications. Snapshots in XtremIO are designed by EMC to be very space-efficient and to enable the instant availability of copies of a database.⁵

⁴ xtremio.com/data-reduction

⁵ xtremio.com/snapshot

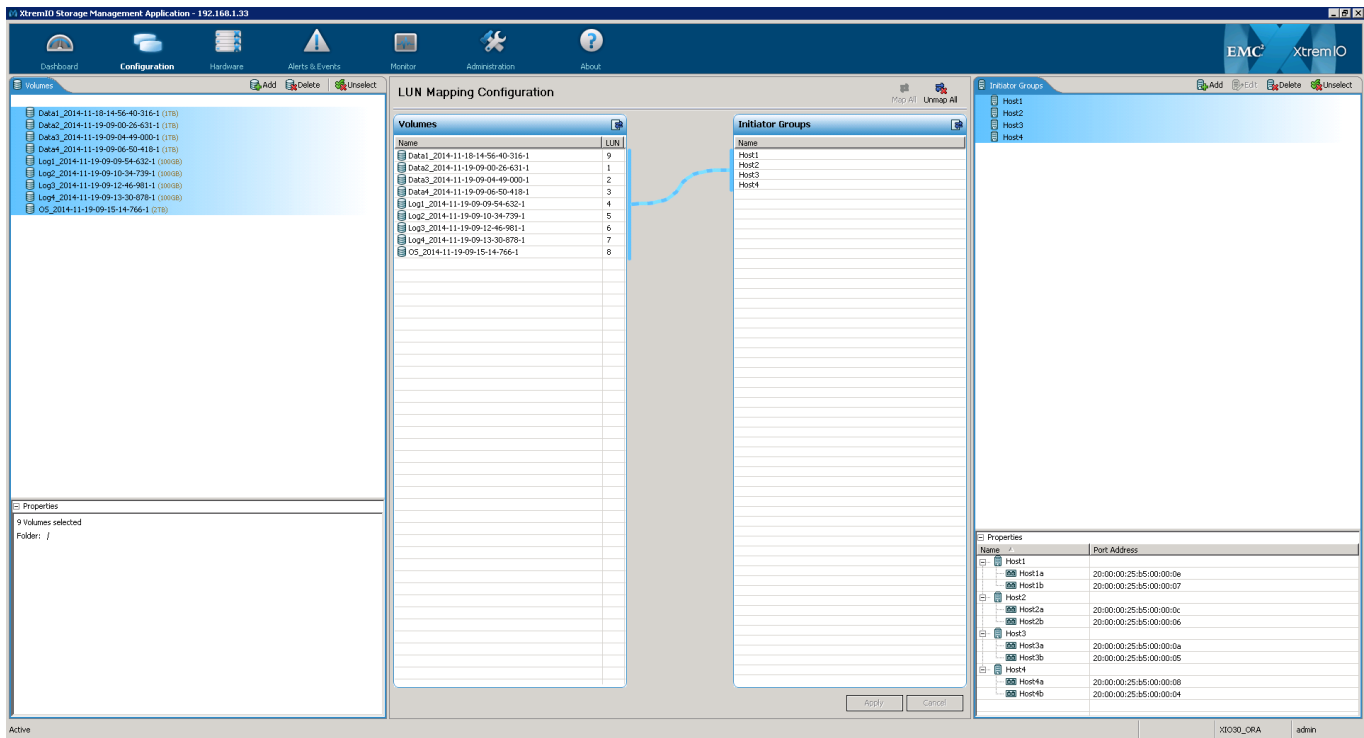


Figure 5: XtremIO Storage Management Application—provisioning LUNs.

Business-critical databases and applications typically require a high degree of availability and integration with virtualized solutions. XtremIO’s N-way active-active controller architecture is created to be extremely reliable and use redundancy wherever possible.⁶ XtremIO Data Protection (XDP) uses N+2 protection on every X-Brick, and seeks to provide protection against multiple SSD failures.⁷ XtremIO integrates with VMware by providing a Virtual Storage Integrator (VSI) and VAAI plug-ins at no charge.

Using EMC protection technologies like RecoverPoint®, VPLEX®, or Data Domain® alongside XtremIO offers integrations with these protection solutions, extending the availability of your data. The performance, space-saving technologies, and built-in data protection and integration with other EMC protection technologies can give your business a powerful option for virtualization storage. To learn more, visit www.emc.com/XtremIO, and for the latest technical papers visit: www.emc.com/everythingoracle.

About VMware vSphere

VMware vSphere is the flagship platform of virtualized hardware from VMware. vSphere allows companies to virtualize their server, storage, and networking resources, achieving a consolidation ratio in some cases greater than 15:1. Features included in

⁶ xtremio.com/high-availability

⁷ xtremio.com/data-protection-2

vSphere include automated management and dynamic resource allocation. To learn more about VMware vSphere, visit www.vmware.com/products/vsphere/overview.html.

According to VMware, vCenter Server™ “provides a centralized platform for managing VMware vSphere environments.” vCenter™ offers visibility into a vSphere infrastructure to analyze and fix issues. Features such as automated load balancing and out-of-the-box automation workflows create automated proactive management. vCenter can work with third-party solutions to extend its virtualization capabilities. For more information, visit www.vmware.com/products/vcenter-server/.

About the Cisco UCS B200 M3 Blade Server

The Cisco UCS® B200 M3 Blade Server is an enterprise-class blade server powered by the new Intel® Xeon® processor E5-2600 series to deliver high performance and outstanding I/O throughput for your applications. Supporting up to 384 GB of RAM with 24 DIMM slots, the Cisco UCS B200 M3 has expansive memory capabilities with Cisco® Extended Memory Technology to support your heavy workloads. The B200 M3 blade will support a configuration of up to 768 GB when 32GB DIMMs are available later this year.

According to Cisco, in addition to performance benefits, Cisco UCS “has the architectural advantage of not having to power and cool switches in each blade chassis. Having a larger power budget available for blades enables Cisco to design uncompromised expandability and capabilities in its blade servers...”⁸ thus saving enterprises power costs while offering competitive performance.

The Cisco Nexus® 5548UP switch offers unified ports that support Ethernet connections as well as Fibre Channel and Fibre Channel over Ethernet connections. The flexibility of the switch and its ports provide a unified networking system for both LAN and storage connections. With 32 ports, support for any Ethernet-based transport, and storage traffic including iSCSI, NAS, FC, RoE, and iBoE, the Nexus 5548UP switch provides excellent consolidation, scalability, and support for evolving technologies all while maintaining a small footprint in your datacenter.⁹

To learn more, visit www.cisco.com/en/US/products/ps12288/index.html.

CONCLUSION

Oracle single instance database VMs need plenty of storage capacity and performance to handle increased workload demands placed on them by users. Whether your organization uses DBaaS or traditional Oracle 12c instances, consider the reliable performance and scaling flexibility that the EMC XtremIO storage array can offer. We found IOPS levels stayed consistent as we scaled up to eight Oracle single instance VMs

⁸ www.cisco.com/c/en/us/products/servers-unified-computing/ucs-b200-m3-blade-server/index.html

⁹ www.cisco.com/c/en/us/products/switches/nexus-5548up-switch/index.html

and scaled by an average of 14,700 IOPS for each VM (totaling 118,067). In addition, we found that the inline deduplication, compression, and thin provisioning capabilities on the XtremIO array resulted in an overall efficiency ratio of 51 to 1 and a data reduction ratio of 14.6 to 1. With this level of consistent performance, users can expect great performance to meet high demand for IOPS in a DBaaS environment.

APPENDIX A – TESTING CONFIGURATIONS AND BENCHMARK

About our database configuration

About our Oracle configuration

We created a new VM, installed Oracle Enterprise Linux® on the VM, and configured the VM with 32GB RAM and 8 vCPUs. Each VM had a 100GB virtual disk, on which we installed the Oracle OS and database software. We then added four 50GB VMDKs to hold the database data files, and four 5GB VMDKs to store the database log files.

We made the necessary networking adjustments and prepared the VM for Oracle installation prerequisites (see [Appendix B](#)). In addition, we configured other OS settings, such as stopping unneeded services, configuring NTP, setting the tuned profile to enterprise-storage, and setting the system's kernel parameters in `sysctl.conf`. We also configured the VM to reserve huge pages for Oracle.

We installed Oracle Grid Infrastructure 12c for Linux x86-64 on the VMs on all nodes. We created additional Oracle ASM disk groups, and then installed and configured the Oracle database. For each database instance, we set the database to use `large_pages` only. We also created a new 32GB UNDO tablespace, and two, 4k blocksize REDO log tablespaces.

Finally, we used the SLOB 2.2 tool to generate the initial database schema. We used a benchmark scale of 150,000 in SLOB's workload generator to create approximately 145GB of data, and we configured SLOB to create the tables and indices. For details, including the SLOB configuration file, see [Appendix C](#) and [Appendix D](#).

About the SLOB 2.2 benchmark

The Silly Little Oracle Benchmark (SLOB) can assess Oracle random physical I/O capability on a given platform in preparation for potential OLTP/ERP-style workloads by measuring IOPS capacity. The benchmark helps evaluate performance of server hardware and software, storage system hardware and firmware, and storage networking hardware and firmware.

SLOB contains simple PL/SQL and offers the ability to test the following:

1. Oracle logical read (SGA buffer gets) scaling
2. Physical random single-block reads (db file sequential read)
3. Random single block writes (DBWR flushing capacity)
4. Extreme REDO logging I/O

SLOB is free of application contention yet is an SGA-intensive benchmark. According to SLOB's creator Kevin Closson, SLOB can also offer more than testing IOPS capability such as studying host characteristics via NUMA and processor threading. For more information on SLOB, links to information on version 2.2, and links to download the benchmark, visit kevinclosson.net/2012/02/06/introducing-slob-the-silly-little-oracle-benchmark/.

APPENDIX B – SYSTEM CONFIGURATION INFORMATION

Figure 6 provides detailed configuration information for the XtremIO storage array.

Storage array	EMC XtremIO
Number of storage shelves	1 × 24 disks enclosure
Number of storage controllers	2
Controller OS	XtremIO OS 2.4.1-11
Disk model number	25 × Hitachi® HUSML404
Disk size (GB)	400
Disk type	SAS SSD

Figure 6: Detailed configuration information for the storage array.

Figure 7 provides detailed configuration information for the test systems.

System	4× Cisco UCS B200 M3 server
General	
Number of processor packages	2
Number of cores per processor	8
Number of hardware threads per core	2
System power management policy	Default
CPU	
Vendor	Intel
Name	Xeon
Model number	E5-2680
Stepping	7
Socket type	LGA2011
Core frequency (GHz)	2.7
Bus frequency	8.00 GT/s
L1 cache	32 KB + 32 KB
L2 cache	256 KB per core
L3 cache	20 MB
Platform	
Vendor and model number	Cisco UCS B200 M3
Motherboard model number	Cisco FCH1607GV4
BIOS name and version	Cisco B200M3.2.1.1a.0.111220131105
BIOS settings	Default

System	4× Cisco UCS B200 M3 server
Memory modules	
Total RAM in system (GB)	320
Vendor and model number	16 × Cisco UCS-MR-1X162RY-A16, 8 × Cisco UCS-MR-1X082RY-A
Type	PC3-12800
Speed (MHz)	1,600
Speed running in the system (MHz)	1,333
Size (GB)	(16×) 16, (8×) 8
Number of RAM module(s)	24 (16 + 8)
Chip organization	Double-sided
Rank	Dual
Hypervisor	
Name	VMware vSphere 5.5.0
Build number	2026576
Language	English
RAID controller	
Vendor and model number	LSI® MegaRAID® SAS 2004
Firmware version	20.10.1-0100
Hard drives	
Vendor and model number	Seagate® A03-D146GC2
Number of drives	2
Size (GB)	146
RPM	15,000
Type	SAS
Converged I/O adapters	
Vendor and model number	Cisco UCSB-MLOM-40G-01, Cisco UCS-VIC-M82-8P
Type	mLOM, Mezzanine
Virtual machine operating system	
Name	Oracle Enterprise Linux Release 6.6
Kernel	3.8.13-44.1.5.el6uek.x86_64
Language	English
Database software	Oracle Database 12c Build 12.1.0.1.0
Database benchmarks	
Benchmark 1	SLOB v2.2

Figure 7: Configuration information for the systems used in our tests.

Figure 8 provides the firmware information for the Cisco hardware we used in our tests.

Cisco hardware	Firmware version
UCS 5108 chassis	
UCS Manager	2.2(3b)
UCS 2208XP IO Module 1 & 2	2.2(3b)
UCS 6248UP Fabric Interconnect 1 & 2	5.2(3)N2(2.23b)

UCS B200 M3 blades	
BIOS	B200M3.2.2.3.0.080820141339
CIMC Controller	2.2(3b)
UCS VIC 1240	4.0(1b)

Figure 8: Firmware information for the Cisco hardware used in our tests.

APPENDIX C – HOW WE TESTED

Oracle Linux 6.5 and Oracle Database 12c

We configured each VM with Oracle Linux 6.5 and Oracle Database 12c. For each VM, we first installed Oracle Grid Infrastructure and utilized Oracle Automatic Storage Management to create two four-disk ASM groups for data and redo logs. We then installed and configured Oracle Database 12c utilizing these ASM disk groups for our database files.

Installing Oracle Linux 6.5

1. Insert the Oracle Linux 6.5 DVD into the server, and boot to it.
2. Select Install or upgrade an existing system.
3. If you are unsure of the fidelity of the installation disk, select OK to test the installation media; otherwise, select Skip.
4. In the opening splash screen, select Next.
5. Choose the language you wish to use, and click Next.
6. Select the keyboard layout, and click Next.
7. Select Basic Storage Devices, and click Next.
8. Select Fresh Installation, and click Next.
9. Insert the hostname, and select Configure Network.
10. In the Network Connections menu, configure network connections.
11. After configuring the network connections, click Close.
12. Click Next.
13. Select the nearest city in your time zone, and click Next.
14. Enter the root password, and click Next.
15. Select Use All Space, and click Next.
16. When the installation prompts you to confirm that you are writing changes to the disk, select Write changes to disk.
17. Select Software Basic Server, and click Next. Oracle Linux installation begins.
18. When the installation completes, select Reboot to restart the server.

Installing VMware Tools

1. Install guest tools or agents.
2. (VMware only) Install VMware Tools on the guest:
 - Right-click the VM in the Web Client, and select Install/Upgrade VMware Tools.
 - Log on to the guest as `root`.
 - Mount the CD ROM device:

```
# mount /dev/cdrom /mnt
```
 - Untar VMware Tools into a temporary directory:

```
# tar -C /tmp -zxf /mnt/VMwareTools-9.4.0-1280544.tar.gz
```
 - Run the install script and accept the defaults:

```
# /tmp/vmware-tools-distrib/vmware-install.pl
```
 - Follow the prompts to configure and install VMware tools.
 - The installer will automatically load the NIC drivers, create a new `initrd`, and unmount the CD.
 - Reboot the VM.

Initial configuration tasks

Complete the following steps to provide the functionality that Oracle Database requires. We performed all of these tasks as root.

1. Disable firewall services. In the command line (as root), type:

```
# service iptables stop
# chkconfig iptables off
# service ip6tables stop
# chkconfig ip6tables off
```
2. Set SELinux:

```
# vi /etc/selinux/config
SELINUX=permissive
```
3. Modify `/etc/hosts` to include the IP address of the internal IP and the hostname.
4. Edit `90-nproc.conf`:

```
# vim /etc/security/limits.d/90-nproc.conf
```

Change this:

```
* soft nproc 1024
```

To this:

```
* - nproc 16384
```
5. Install 12c RPM packages, resolve package dependencies, and modify kernel parameters:

```
# yum install oracle-rdbms-server-12cR1-preinstall -y
```
6. Install automatic system tuning for database storage through yum:

```
# yum install tuned
# chkconfig tuned on
# tuned-adm profile enterprise-storage
```
7. Using yum, install the following prerequisite packages for Oracle Database:

```
# yum install elfutils-libelf-devel
# yum install xhost
# yum install unixODBC
# yum install unixODBC-devel
# yum install oracleasm-support oracleasm lib oracleasm
```
8. Create the oracle user account and groups and password:

```
# groupadd -g 1003 oper
# groupadd -g 1004 asmadmin
# groupadd -g 1005 asmdba
# groupadd -g 1006 asmoper
# usermod -G dba,oper,asmadmin,asmdba,asmoper oracle
# passwd oracle
```
9. Create the `/u01` directory for Oracle inventory and software and give it to the oracle user:

```
# mkdir -p /u01/app/oracle/product/12.1.0/grid_1
# mkdir -p /u01/app/oracle/product/12.1.0/dbhome_1
# chown -R oracle:oinstall /u01
# chmod -R 775 /u01
```
10. Edit bash profiles to set up user environments:

```
# vim /home/oracle/.bash_profile
```

```

# Oracle Settings
export TMP=/tmp
export TMPDIR=$TMP
export ORACLE_HOSTNAME=localhost.localdomain
export ORACLE_BASE=/u01/app/oracle
export GRID_HOME=$ORACLE_BASE/product/12.1.0/grid_1
export DB_HOME=$ORACLE_BASE/product/12.1.0/dbhome_1
export ORACLE_HOME=$DB_HOME
export ORACLE_SID=orcl
export ORACLE_TERM=xterm
export BASE_PATH=/usr/sbin:$PATH
export PATH=$ORACLE_HOME/bin:$BASE_PATH
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/lib:/usr/lib
export CLASSPATH=$ORACLE_HOME/JRE:$ORACLE_HOME/jlib:$ORACLE_HOME/rdbms/jlib
alias grid_env='. /home/oracle/grid_env'
alias db_env='. /home/oracle/db_env'

# vim /home/oracle/grid_env

export ORACLE_SID=+ASM1
export ORACLE_HOME=$GRID_HOME
export PATH=$ORACLE_HOME/bin:$BASE_PATH
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/lib:/usr/lib
export CLASSPATH=$ORACLE_HOME/JRE:$ORACLE_HOME/jlib:$ORACLE_HOME/rdbms/jlib

# vim /home/oracle/db_env

export ORACLE_SID=orcl
export ORACLE_HOME=$DB_HOME
export PATH=$ORACLE_HOME/bin:$BASE_PATH
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/lib:/usr/lib
export CLASSPATH=$ORACLE_HOME/JRE:$ORACLE_HOME/jlib:$ORACLE_HOME/rdbms/jlib

```

Adding the four data and four log drives to the VM and modifying the vmx file

1. Power off the VM.
2. Right-click the VM in the vSphere Web Client, and choose Edit Settings...
3. Click on the VM Options tab, and expand the Advanced menu option.
4. Choose Edit Configuration...
5. Click Add Row, and enter `disk.EnableUUID` in the parameter field and `TRUE` in the value field.
6. Go back to the Virtual Hardware tab.
7. Click the drop-down menu for New device, and choose New Hard Disk.
8. Name the Hard Disk and choose the size that you want it to be.
9. Repeat steps 7 and 8 for all remaining drives.
10. Click OK.
11. Power the VM back on.

Configuring disks for ASM

1. For each of the eight shared disks, create a GPT label, and create one partition. For example, see the following shell script:

```
for disk in sdb sdc sdd sde sdf sdg sdh sdi; do
    parted /dev/$disk mklabel gpt
    parted /dev/$disk mkpart primary "1 -1"
done
```

2. If desired, label the disk's partition with its Oracle function. For example:

```
parted /dev/sdc name 1 DATA1
parted /dev/sdd name 1 DATA2
parted /dev/sde name 1 DATA3
parted /dev/sdf name 1 DATA4
parted /dev/sdg name 1 LOG1
parted /dev/sdh name 1 LOG2
parted /dev/sdi name 1 LOG3
parted /dev/sdi name 1 LOG4
```

3. Initialize Oracle ASM on each server by executing the following commands as root on each node.

```
oracleasm init
oracleasm configure -e -u grid -g oinstall -s y -x sda
```

4. Label each shared disk-partition with an appropriate ASM name. For example, following the OS partition names created above, execute the following commands on one system:

```
oracleasm createdisk DATA1 /dev/sdc1
oracleasm createdisk DATA2 /dev/sdd1
oracleasm createdisk DATA3 /dev/sde1
oracleasm createdisk DATA4 /dev/sdf1
oracleasm createdisk LOG1 /dev/sdg1
oracleasm createdisk LOG2 /dev/sdh1
oracleasm createdisk LOG3 /dev/sdi1
oracleasm createdisk LOG4 /dev/sdj1
```

5. On each server, scan the disks to make the disks immediately available to Oracle ASM.

```
oracleasm scandisks
oracleasm listdisks
```

Installing Oracle Grid Infrastructure 12c

1. Log in as the oracle user.
2. Unzip linuxamd64_12c_grid_1of2.zip and linuxamd64_12c_grid_2of2.zip
3. Open a terminal to the unzipped database directory.
4. Type `grid_env` to set the Oracle grid environment.
5. To start the installer, type `./runInstaller`
6. At the Updates screen, select Skip updates.
7. In the Select Installation Option screen, select Install and Configure Grid Infrastructure for a Standalone Server, and click Next.
8. Choose the language, and click Next.
9. In the Create ASM Disk Group screen, choose the Disk Group Name, and change redundancy to External.
10. Select the four disks that you are planning to use for the database, and click Next.
11. In the Specify ASM Password screen, choose Use same password for these accounts, write the passwords for the ASM users, and click Next.

12. Leave the default Operating System Groups, and click Next.
13. Leave the default installation, and click Next.
14. Leave the default inventory location, and click Next.
15. Under Root script execution, select Automatically run configuration scripts and enter root credentials.
16. In the Prerequisite Checks screen, make sure that there are no errors.
17. In the Summary screen, verify that everything is correct, and click Finish to install Oracle Grid Infrastructure.
18. At one point during the installation, the installation prompts you to execute two configuration scripts as root. Follow the instructions to run the scripts.
19. At the Finish screen, click Close.
20. To run the ASM Configuration Assistant, type `asmca`.
21. In the ASM Configuration Assistant, click Create.
22. In the Create Disk Group window, name the new disk group `log`, choose redundancy External (None), and select the four disks for redo logs.
23. Click Advanced Options, and type 12.1.0.0.0 in ASM Compatibility and Database Compatibility. Click OK.
24. Right-click the DATA drive, and choose Edit Attributes. Make sure both ASM and Database Compatibility fields list 12.1.0.0.0, and click OK.
25. Exit the ASM Configuration Assistant.

Installing Oracle Database 12c

1. Unzip `linuxamd64_12c_database_1_of_2.zip` and `linuxamd64_12c_database_2_of_2.zip`.
2. Open a terminal to the unzipped database directory.
3. Type `db_env` to set the Oracle database environment.
4. Run `./runInstaller.sh`.
5. Wait for the GUI installer loads.
6. On the Configure Security Updates screen, enter the credentials for My Oracle Support. If you do not have an account, uncheck the box I wish to receive security updates via My Oracle Support, and click Next.
7. At the warning, click Yes.
8. On the Download Software Updates screen, enter the desired update option, and click Next.
9. On the Select Installation Option screen, select Install database software only, and click Next.
10. On the Grid Installation Options screen, select Single instance database installation, and click Next.
11. On the Select Product Languages screen, leave the default setting of English, and click Next.
12. On the Select Database Edition screen, select Enterprise Edition, and click Next.
13. On the Specify Installation Location, leave the defaults, and click Next.
14. On the Create Inventory screen, leave the default settings, and click Next.
15. On the Privileged Operating System groups screen, keep the defaults, and click Next.
16. Allow the prerequisite checker to complete.
17. On the Summary screen, click Install.
18. Once the Execute Configuration scripts prompt appears, ssh into the server as `root`, and run the following command:

```
# /home/oracle/app/oracle/product/12.1.0/dbhome_1/root.sh
```
19. Return to the prompt, and click OK.
20. Once the installer completes, click Close.

Adding hugepages

1. In Putty, edit the `sysctl.conf` file (`vi /etc/sysctl.conf`) by adding the following lines:

```
vm.nr_hugepages = 13312
vm.hugetlb_shm_group = 1001
```

Creating and configuring the database

1. Using Putty with X11 forwarding enabled, SSH to the VM.
2. Type `dbca`, and press Enter to open the Database configuration assistant.
3. At the Database Operation screen, select Create Database, and click Next.
4. Under Creation Mode, select Advanced Mode, and click Next.
5. At the Select Template screen, select General Purpose or Transaction Processing. Click Next.
6. Enter a Global database name and the appropriate SID.
7. At the Management Options screen, select Configure Enterprise Manager (EM) Database Express. Click Next.
8. At the Database Credentials screen, select Use the Same Administrative Password for All Accounts. Enter a password, and click Next.
9. At the Network Configuration screen, click Next.
10. At the Storage Configuration screen, select Automatic Storage Management, and select `+DATA` as the Database.
11. At the Database Options screen, click Next.
12. At the Initialization Parameters screen, click use Automatic Memory Management.
13. At the Creation Options screen, select Create Database, and click Next.
14. At the summary screen, click Finish.
15. Close the Database Configuration Assistant.
16. In a Web browser, browse to `https://vm.ip.address:5500/em` to open the database manager.
17. Log in as system with the password you specified.
18. Go to Storage→Tablespaces.
19. Click Create.
20. Enter `SLOB` as the Name, and check the Set As Default box. Click OK.
21. Go to Storage→Redo Log Groups.
22. Click Actions→Switch file... until you get one of the groups to go inactive.
23. Highlight the inactive group, and click Actions→Drop group.
24. In Putty, enter `sqlplus` and type `ALTER DATABASE ADD LOGFILE GROUP 3 ('+logs/orcl/onlinelog/redox.log') SIZE 9G BLOCKSIZE 4k; where x is the number of the log file (1, 2, etc.).`
25. Repeat steps 21 – 24 until all default logs are dropped, and two new ones exist.
26. Click on Configuration→Initialization Parameters
27. Change the `use_large_pages` parameter to Only.

Installing SLOB and populating the database

1. Download the SLOB kit from www.kevinclosson.net/slob/.
2. Copy and untar the files to `/home/oracle/dev/`.
3. Edit the `slob.conf` file to match [Appendix D](#).
4. Type `./setup.sh SLOB 128` to start the data population to the SLOB tablespace we created earlier.
5. When the setup is complete, the database is populated.

Running the performance test

1. Once the first VM is set up and properly populated with data, clone out seven more VMs for a total of eight.
2. We used a round-robin approach on the hosts so that VM1 and VM5 were on host 1, VM2 and VM6 were on host 2, etc.
3. Once all VMs have cloned out, make sure they have unique IPs, and edit `/etc/hosts` to ensure the host IP matches.
4. Start `esxtop` on all four hosts.
5. With all VMs turned on, start the test on the first VM (`./runit.sh 128`), and start a 5 minute timer.
6. When the timer goes off, start the test on the second VM, and start a 5 minute timer.
7. Repeat steps 5 and 6 until all VMs have started the test.

Running the cloning test

1. Using the command line utility for the XIO array, we reset the storage for a clean start:

```
stop-cluster
restart-xms
start-cluster
```
2. Once the storage was back online, we created a single 5TB LUN, and presented the LUN to the four Cisco blades.
3. Using the same VM template as we used for the previous test, we cloned one VM onto the storage from a remote location.
4. Using the EMC VSI plug-in, we cloned 7 more VMs from the first VM sitting on the storage.
5. Once the cloning was finished, we took a screenshot to capture the deduplication, compression, and efficiency data as shown in the EMC console.
6. We performed this test 3 times, and took the median total efficiency score for our final run.

APPENDIX D – BENCHMARK PARAMETERS

We used the following slob.conf parameter settings.

```
UPDATE_PCT=20
RUN_TIME=7200
WORK_LOOP=0
SCALE=150000
WORK_UNIT=16
REDO_STRESS=LITE
LOAD_PARALLEL_DEGREE=16
SHARED_DATA_MODULUS=0

# Settings for SQL*Net connectivity:
#ADMIN_SQLNET_SERVICE=slob
#SQLNET_SERVICE_BASE=slob
#SQLNET_SERVICE_MAX=2
#SYSDBA_PASSWD=change_on_install

#####
#### Advanced settings:
DO_UPDATE_HOTSPOT=FALSE
HOTSPOT_PCT=10

THINK_TM_MODULUS=5
THINK_TM_MIN=.7
THINK_TM_MAX=.7
#####

export UPDATE_PCT RUN_TIME WORK_LOOP SCALE WORK_UNIT LOAD_PARALLEL_DEGREE
REDO_STRESS SHARED_DATA_MODULUS
export DO_UPDATE_HOTSPOT HOTSPOT_PCT THINK_TM_MODULUS THINK_TM_MIN THINK_TM_MAX
```

ABOUT PRINCIPLED TECHNOLOGIES



Principled Technologies, Inc.
1007 Slater Road, Suite 300
Durham, NC, 27703
www.principledtechnologies.com

We provide industry-leading technology assessment and fact-based marketing services. We bring to every assignment extensive experience with and expertise in all aspects of technology testing and analysis, from researching new technologies, to developing new methodologies, to testing with existing and new tools.

When the assessment is complete, we know how to present the results to a broad range of target audiences. We provide our clients with the materials they need, from market-focused data to use in their own collateral to custom sales aids, such as test reports, performance assessments, and white papers. Every document reflects the results of our trusted independent analysis.

We provide customized services that focus on our clients' individual requirements. Whether the technology involves hardware, software, Web sites, or services, we offer the experience, expertise, and tools to help our clients assess how it will fare against its competition, its performance, its market readiness, and its quality and reliability.

Our founders, Mark L. Van Name and Bill Catchings, have worked together in technology assessment for over 20 years. As journalists, they published over a thousand articles on a wide array of technology subjects. They created and led the Ziff-Davis Benchmark Operation, which developed such industry-standard benchmarks as Ziff Davis Media's Winstone and WebBench. They founded and led eTesting Labs, and after the acquisition of that company by Lionbridge Technologies were the head and CTO of VeriTest.

Principled Technologies is a registered trademark of Principled Technologies, Inc.
All other product names are the trademarks of their respective owners.

Disclaimer of Warranties; Limitation of Liability:

PRINCIPLED TECHNOLOGIES, INC. HAS MADE REASONABLE EFFORTS TO ENSURE THE ACCURACY AND VALIDITY OF ITS TESTING, HOWEVER, PRINCIPLED TECHNOLOGIES, INC. SPECIFICALLY DISCLAIMS ANY WARRANTY, EXPRESSED OR IMPLIED, RELATING TO THE TEST RESULTS AND ANALYSIS, THEIR ACCURACY, COMPLETENESS OR QUALITY, INCLUDING ANY IMPLIED WARRANTY OF FITNESS FOR ANY PARTICULAR PURPOSE. ALL PERSONS OR ENTITIES RELYING ON THE RESULTS OF ANY TESTING DO SO AT THEIR OWN RISK, AND AGREE THAT PRINCIPLED TECHNOLOGIES, INC., ITS EMPLOYEES AND ITS SUBCONTRACTORS SHALL HAVE NO LIABILITY WHATSOEVER FROM ANY CLAIM OF LOSS OR DAMAGE ON ACCOUNT OF ANY ALLEGED ERROR OR DEFECT IN ANY TESTING PROCEDURE OR RESULT.

IN NO EVENT SHALL PRINCIPLED TECHNOLOGIES, INC. BE LIABLE FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH ITS TESTING, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL PRINCIPLED TECHNOLOGIES, INC.'S LIABILITY, INCLUDING FOR DIRECT DAMAGES, EXCEED THE AMOUNTS PAID IN CONNECTION WITH PRINCIPLED TECHNOLOGIES, INC.'S TESTING. CUSTOMER'S SOLE AND EXCLUSIVE REMEDIES ARE AS SET FORTH HEREIN.
