PT

Manage transactional and data mart loads with superior performance and high availability

The Dell EMC VMAX 250F All Flash storage array supported database workloads better than the HPE 3PAR 8450 storage array

Dell EMC VMAX 250F

It's not enough to make the transaction process as fast and effortless as possible; the speed of business also demands a backend storage solution that won't bog down when faced with large data mart transfers or let unplanned downtime interfere with business as usual. All-Flash storage arrays from Dell EMC™ and HPE promise to help companies avoid those pitfalls. But how well do they live up to that promise?

We set up and tested solutions from both vendors and found that the Dell EMC VMAX™ 250F All Flash storage array paired with PowerEdge™ servers came out ahead of the HPE 3PAR 8450 storage array backed by ProLiant servers in two key areas:

- The VMAX 250F processed transactional and data mart loading at the same time with minimal impact to wait times or database performance.
- When faced with interruptions in access to local storage, the database host seamlessly redirected all I/O to the remote VMAX 250F via SRDF/Metro with no interruption of service or downtime.

This report shows you how each system can help or hinder data access during backups and unplanned downtime.
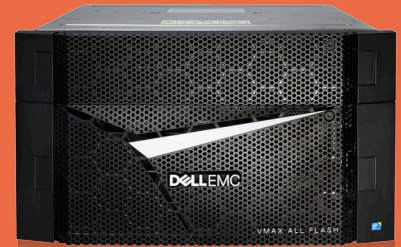
Latency 1ms or less
keeps wait times unnoticeable

Minimal impact
during data mart load

No downtime
with loss of local
storage connectivity

Support customers and maintenance situations at the same time

Companies routinely gather information from various offices or departments and store them in one place so it's easier to keep track of sales, create financial reports, check expenses, and perform Big Data analysis. As the sheer volume of useable information grows, so does the strain on infrastructure. These initiatives can grind all other business to a halt.

No performance loss during load times means IT doesn't have to wait to gather important information in one place. So, you can access numbers that are up-to-the-minute instead of waiting until it's convenient.

The purpose of data marts

Data marts are a convenient way to move chunks of data from multiple departments to one centralized storage location that's easy to access. Businesses rely on the smooth flow of data mart information for reporting, analysis, trending, presentations, and database backups.

We started our comparison of the Dell EMC and HPE All-Flash storage solutions with a performance evaluation. We enabled compression on the arrays and ran the same two Oracle® Database 12c transactional workloads created by Silly Little Oracle Benchmark (SLOB) on both solutions. Then, we added a data mart load into the mix using a VM running Microsoft® SQL Server® 2016 pushing data from an external source onto the target array. Our experts looked at two things during this process: the average number of input/output operations per second (IOPS) each solution handled and storage latency before and during the data mart load.

Storage fabric using Brocade® Gen 6 hardware

In our tests, we used Connectrix® DS-6620B switches, built on Brocade Gen 6 hardware, known as the Dell EMC Connectrix B-Series Gen 6 Fibre Channel by Brocade. The Connectrix B Series provides out-of-the box tools for SAN monitoring, management, and diagnostics that simplify administration and troubleshooting for administrators.
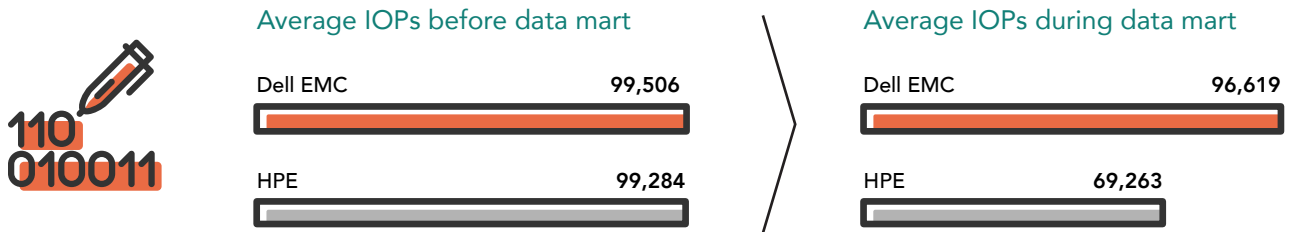
Brocade Fibre offers Brocade Fabric Vision® Technology, which can provide further visibility into the storage network with monitoring and diagnostic tools. With Monitoring and Alerting Policy Suite (MAPS), admins can proactively monitor the health of all connected storage using policy-based monitoring.

Brocade offers another tool to simplify SAN management for Gen 6 hardware: Connectrix Manager Converged Network Edition (CMCNE). This tool uses an intuitive GUI to help admins automate repetitive tasks and further simplify SAN fabric management in the datacenter.
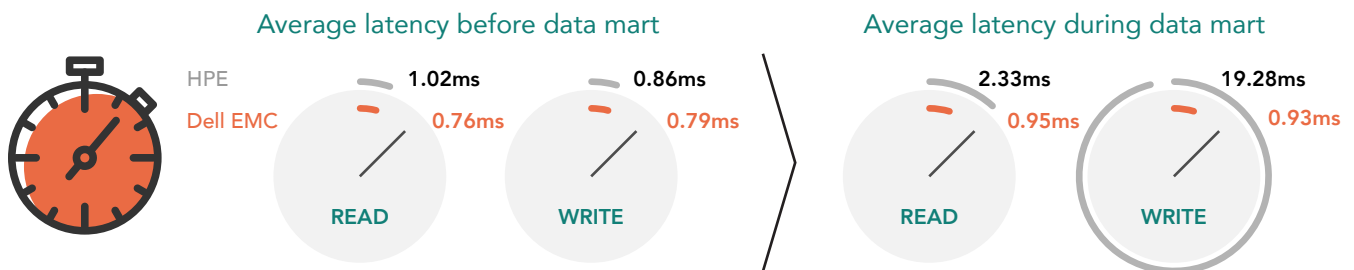
To learn more about Dell EMC Connectrix, visit the Dell EMC Connectrix page.

We found that when we added a large data write load to the transactional loads already running on the VMAX 250F All Flash storage array, performance on those workloads decreased only slightly. Plus, the VMAX solution handled 39.5 percent more IOPS than the 3PAR solution during the data mart load. This is useful because businesses won't have to worry about whether performing an extensive backup or compiling large amounts of data from multiple sources will affect their service.

Average IOPs before data mart

Dell EMC   99,506

HPE   99,284

Average IOPs during data mart

Dell EMC   96,619

HPE   69,263

The HPE 3PAR 8450 solution handled 30 percent less IOPS when we added the data mart load, and latency times also increased dramatically. This step down can negatively affect customer experience: long wait times can wear on users accessing data and frustrate them.

The average storage latency for both reads and writes on the Dell EMC VMAX 250F solution stayed under a millisecond. Plus, the VMAX solution handled reads and writes much faster than the 3PAR solution during the data mart load—up to 145 percent faster reads and 1,973 percent faster writes. This means data access won't slow down even when you schedule maintenance during peak use hours.

Average latency before data mart

HPE

Dell EMC

1.02ms   0.76ms   0.86ms   0.79ms

READ   WRITE

Average latency during data mart

2.33ms   0.95ms   19.28ms   0.93ms

READ   WRITE

By contrast, storage latency for the HPE 3PAR 8450 solution was higher and the HPE storage array experienced lengthy delays when processing reads and writes at the same time—read latency increased 129 percent while write latency increased 2,133 percent. It's hard to ignore delays that long. For a detailed breakdown of the IOPS and storage latency results, see Appendix A.

Prepare for unexpected downtime with reliable failover protection

Databases are vital to organizations, so having a reliable backup plan is crucial. Your datacenter shouldn't stop in its tracks because of a connection failure to an active storage array. Failover technology should affect day-to-day business as little as possible.

The second phase of our comparison of the Dell EMC and HPE All-Flash storage solutions involved remote replication and disaster recovery. For the Dell EMC solution, we used Unisphere for VMAX to set up two VMAX 250F arrays with active-active SRDF/Metro remote replication. This meant that the database I/O was running against the local and remote arrays—with the load split evenly 50/50. For the HPE solution, we set up one 3PAR 8450 array and one 3PAR 8400 array with Remote Copy and Peer Persistence enabled. This meant the 8400 array would only be available if the 8450 failed. Then, we deployed one 1.2 TB Oracle instance configured to run an OLTP workload on each system. Once the Oracle instances were up and running, we initiated a lost host connection on the local arrays.

SRDF/Metro benefits

Load balancing

Redundancy support

Active paths to both sites

No downtime

The entire workload on the VMAX 250F solution continued to run with no downtime following the outage while all I/O shifted immediately to the remote 250F. In contrast, the application workload on the 3PAR solution needed to transfer 100 percent of the I/O to the secondary site and stopped until we restarted the VM. SRDF/Metro is active-active, which ensured consistent data access during our site failure. HPE Peer Persistence is active-passive, so, during our local storage failure, all the paths were inaccessible until the standby paths to the remote array become active or we restored the connection to the local system and failback occurred. This can mean the difference between having consistent data access during a site failure and not. For the IOPS charts from the storage arrays, see Appendix A.

Conclusion

In the end, the Dell EMC VMAX 250F All Flash storage array lived up to its promises better than the HPE 3PAR 8450 storage array did.

We experienced minimal impact to database performance when the VMAX 250F processed transactional and data mart loading at the same time. This is useful whether you're performing extensive backups or compiling large amounts of data from multiple sources.

Plus, the VMAX 250F, using the active/active architecture of SRDF/Metro, seamlessly transferred database I/O to a remote VMAX 250F with no interruption of service or downtime when we initiated a lost host connection on the local arrays. By contrast, the HPE 3PAR 8450 solution, using the active/passive architectue of Remote Copy and Peer Persistence, faced downtime until a failover occurred.

These findings prove that the Dell EMC VMAX 250F All Flash storage array is a good option for businesses looking for performance they can rely on.

To find out more about Dell EMC VMAX All Flash, visit DellEMC.com/VMAX.

On July 15, 2017 we finalized the hardware and software configurations we tested. Updates for current and recently released hardware and software appear often, so unavoidably these configurations may not represent the latest versions available when this report appears. For older systems, we chose configurations representative of typical purchases of those systems. We concluded hands-on testing on August 14, 2017.

Appendix A: Detailed results output

Below are charts detailing the results from our testing.

SLOB and data mart testing

IOPS

Below is a chart showing the total IOPS of both SLOB workloads for each environment over the course of a SLOB and data mart test run. When we introduced the data mart load at 30 minutes, the Dell EMC VMAX 250F array continued to support the SLOB workload while the SLOB workload running on the 3PAR array took a performance hit.

Total IOPS
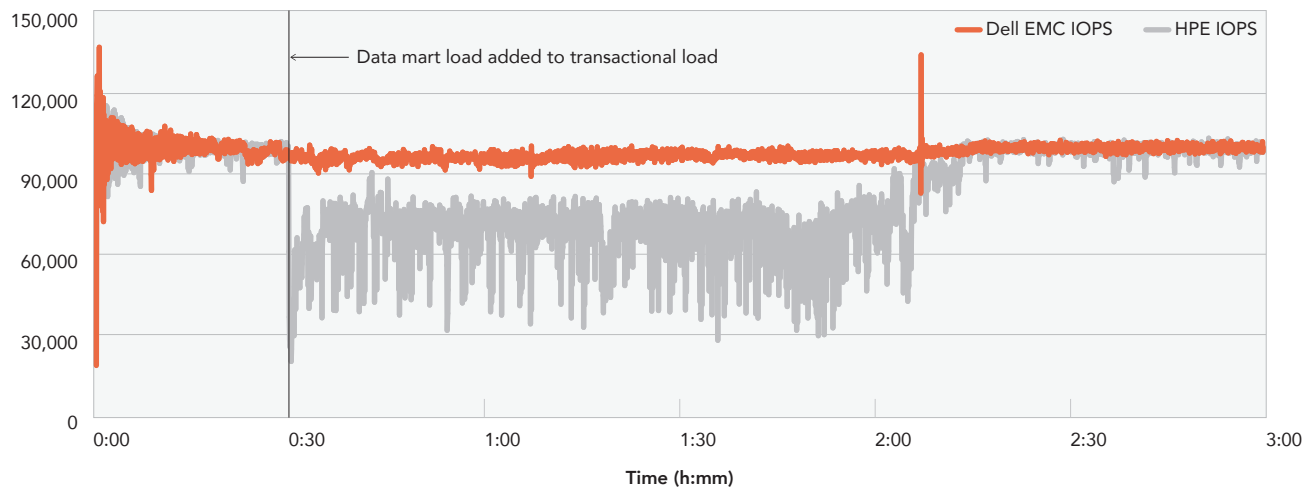
Latency

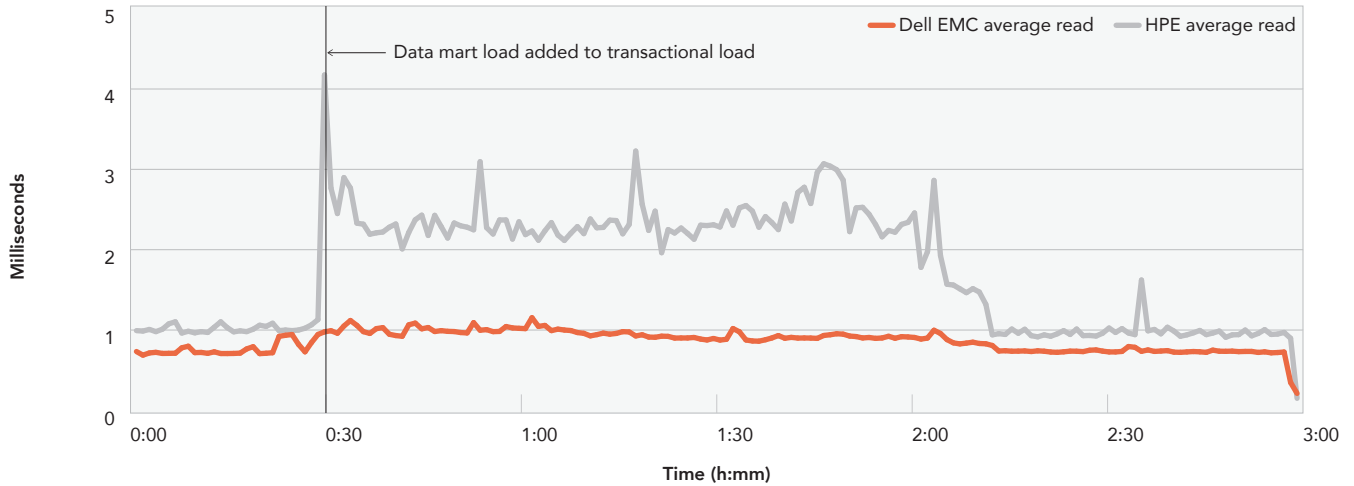Similarly, the charts below show the impact of the data mart load on the storage latency for both reads and writes as seen by the SLOB host servers' storage adapters. Again, the Dell EMC VMAX 250F array held up under the added data mart load. However, the 3PAR 8450 experienced increases in wait times for both reads and especially writes.

Average host read latency

Average host write latency
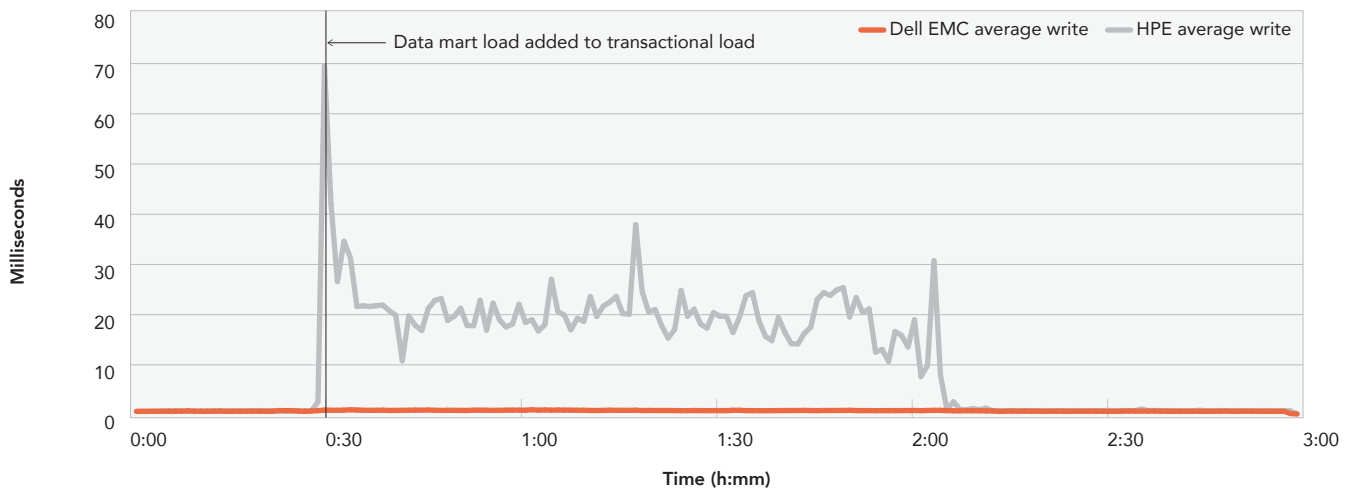
Remote replication testing

SRDF/Metro

Below is a chart showing the storage IOPS from our single Oracle VM during our remote replication testing. For the first 20 minutes, the host perfectly balanced the workload between both sites. When we initiated the storage disconnection, Site B immediately handled 100% of the load. When we restored the connection 40 minutes later, the workload immediately rebalanced on both sites.

SRDF/Metro storage IOPS
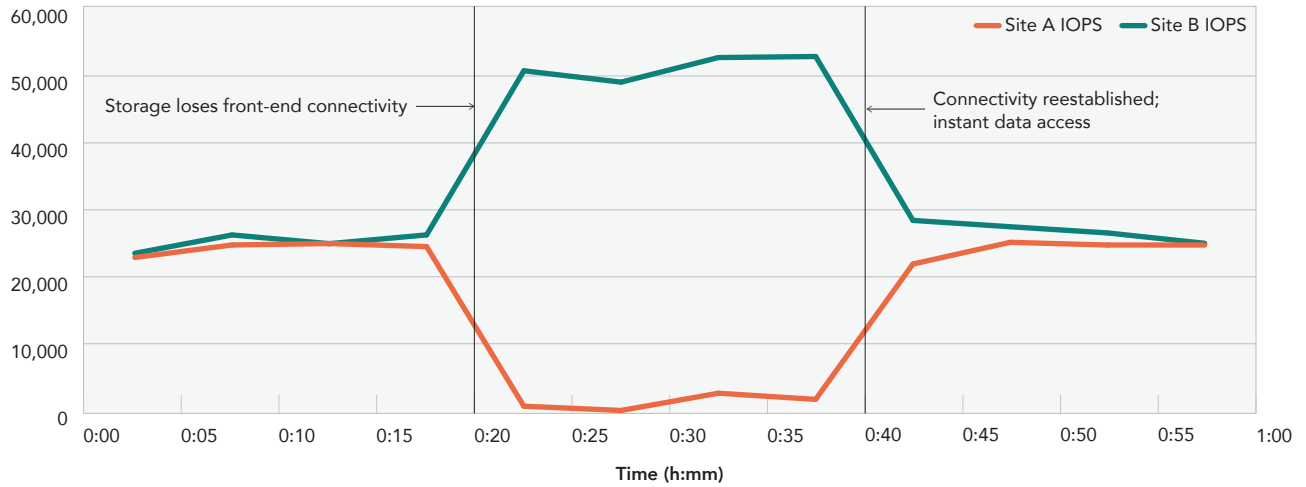
Peer Persistence

In the chart below, we show the storage IOPS for the 3PAR arrays during the remote replication test. For the first 20 minutes, the site A array handled the whole workload while site B remained in standby. When the storage connection failure occurred, the workload immediately crashed. Once the standby paths became active, we were able to restart the VM and kick off the workload. We then restored the site A storage connection and performed a recovery. The recovery back to site A worked with no interruption to the workload.

3PAR Peer Persistence storage IOPS
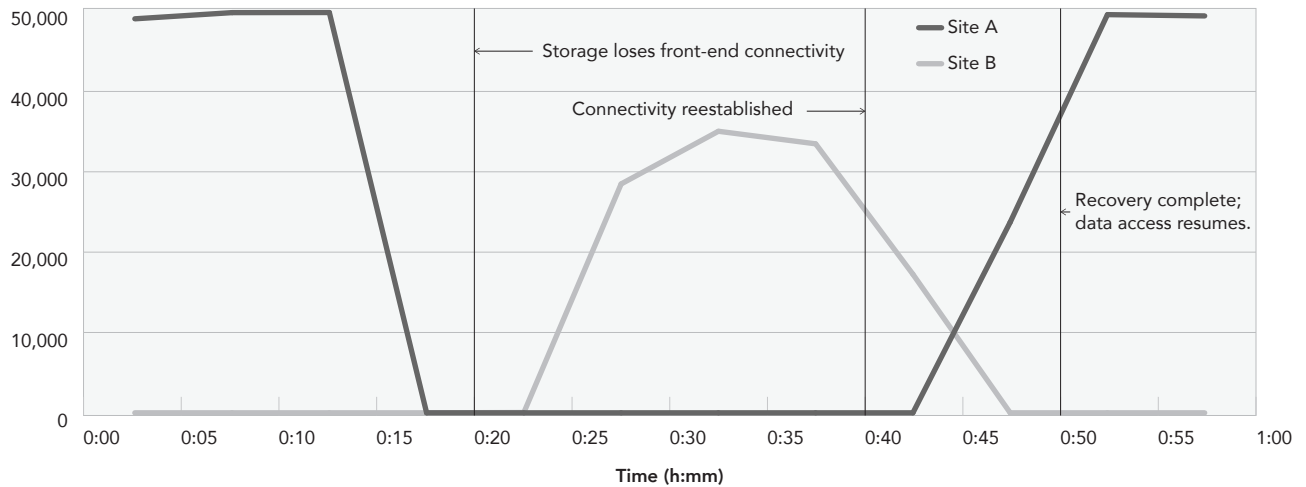
Appendix B: System configuration information

Server configuration information	2x Dell EMC PowerEdge R930	2x HPE ProLiant DL580 Gen9
BIOS name and version	Dell 2.2.0	HP 2.30
Non-default BIOS settings	N/A	Intel Turbo Boost enabled, Virtualization enabled
Operating system name and version/build number	VMware® ESXi™ 6.5	VMware ESXi 6.5
Date of last OS updates/patches applied	05/01/2017	05/01/2017
Power management policy	Performance	Maximum Performance
Processor		
Number of processors	2	2
Vendor and model	Intel® Xeon® E7-4809 v4	Intel Xeon E7-4809 v4
Core count (per processor)	8	8
Core frequency (GHz)	2.1	2.1
Stepping	E0	E0
Memory module(s)		
Total memory in system (GB)	256	256
Number of memory modules	16	16
Vendor and model	Hynix HMA82GRMF8N-UH	Samsung M393A2K40BB1-CRC0Q
Size (GB)	16	16
Type	DDR4	DDR4
Speed (MHz)	2,400	2,400
Speed running in the server (MHz)	2,400	2,400
Storage controller		
Vendor and model	Dell PERC H730P	Smart Array P830i
Cache size (GB)	2	4
Firmware version	25.5.0.0018	4.04_B0
Driver version	6.910.18.00	2.0.10
Local storage		
Number of drives	2	2
Drive vendor and model	Dell ST300MM0006	HP EG0300FBVFL
Drive size (GB)	300	300
Drive information (speed, interface, type)	10K, 6Gb SAS, HDD	10K, 6Gb SAS, HDD
Network adapter #1		
Vendor and model	Broadcom Gigabit Ethernet BCM5720	HP Ethernet 1Gb 4-port 331FLR Adapter
Number and type of ports	4 x 1 GbE	4 x 1GbE
Driver version	4.1.0.0	4.1.0.0

Server configuration information	2x Dell EMC PowerEdge R930	2x HPE ProLiant DL580 Gen9
Network adapter #2		
Vendor and model	Intel 82599	Intel 82599EB
Number and type of ports	2 x 10GbE	2 x 10GbE
Driver version	4.4.1-iov	3.7.13.7.14iov-NAPI
Network adapter #3		
Vendor and model	Emulex LightPulse® LPe31002-M6-D	Emulex LightPulse LPe31002-M6-D
Number and type of ports	2 x 16Gb Fibre	2 x 16Gb Fibre
Driver version	11.1.196.3	11.1.0.6
Cooling fans		
Vendor and model	Nidec®J87TW-A00	Delta Electronics Inc. PFR0912XHE
Number of cooling fans	6	4
Power Supplies		
Vendor and model	Dell D750E-S6	HP DPS-1200SB A
Number of power supplies	4	2
Wattage of each (W)	750	1,200

Storage configuration information	2 x VMAX 250F	HPE 3PAR 8450 (primary)	HPE 3PAR 8400 (secondary)
Controller firmware revision	Hypermax OS 5977.1125.1125	HPE 3PAR OS 3.3.1.215	HPE 3PAR OS 3.3.1.215
Number of storage controllers	2 (1 V-Brick)	2	2
Number of storage shelves	2	2	4
Network ports	32 x 16Gb Fibre Channel	12 x 16Gb Fibre Channel	12 x 16Gb Fibre Channel
RAID Configuration	RAID 5 (3+1)	RAID 5 (3+1)	RAID 5 (3+1)
Drive vendor and model number	Samsung MZ-ILS9600	48 x HPE DOPE1920S5xnNMRI	24 x HPE DOPM3840S5xnNMRI
Drive size (GB)	960	1,920	3,840
Drive information (speed, interface, type)	SAS SSD	SAS SSD	SAS SSD

Appendix C: Testing configurations and benchmarks

Our Dell EMC testbed consisted of three Dell EMC PowerEdge R930 servers. Our HPE testbed consisted of three HPE ProLiant DL580 Gen9 servers. On both testbeds, two servers each hosted one Oracle VM while the third hosted the data mart VM. For the SRDF/Metro and Peer Persistence testing, we simplified the host environment down to a single host and Oracle VM since we were focusing on the behavior of the storage and not a server failover. We also used a two-socket 2U server to host our infrastructure VMs. Each server had a single 1Gb connection to a TOR switch for network traffic. Each server under test had four 16Gb Fibre connections to two Dell EMC DS6620B Fibre Channel switches: one dedicated to the Dell EMC environment and the other for the HPE environment.

The Dell EMC environment used two VMAX 250F arrays: one as the primary site array and the other as the secondary array. Similarly, the HPE side used two HPE 3PAR StoreServ arrays: a 3PAR 8450 as the primary array and a 3PAR 8400 as the secondary. For the SLOB and data mart testing, we used just the primary arrays in each environment. We configured the VMAX 250F with 16 Fibre channel connections, using half of the available ports, while we configured the 3PAR 8450 with 12 Fibre channel connections using all the available ports on the array. For the SRDF/Metro and Peer Persistence testing, we configured each array with eight Fibre channel ports dedicated to the front-end host connections while we dedicated four FC ports to SRDF on the VMAX arrays and Remote Copy on the 3PAR arrays. The VMAX 250F allowed us to allocate director cores directly for certain functions. For the SLOB and data mart testing, we allocated 15 cores to the front-end host ports and then adjusted to 12 for front-end and six for the SRDF ports.

Configuring the Fibre Channel switches

We created Fibre Channel zones according to best practices for each All-Flash array. For the host port connections, we created peer zones containing the host ports as principal members and the storage host ports as Peer Members. For the SRDF connections for VMAX, we configured one peer zone with site A as principal and site B as peer, and then a second peer zone in the opposite direction. For 3PAR, we zoned each Remote Copy port 1-to-1 with its secondary counterpart. We then created different zone configurations for each phase of testing to enable and disable desired ports.

Creating the Fibre Channel zones

1. Log into the Brocade switch GUI.
2. Click Configure→Zone Admin.
3. Click the Zone tab.
4. Click New Peer Zone, and enter a zone name.
5. Select the desired single server port WWNs as principal members and the desired storage WWNs as peer members.
6. Click the right arrow to add the WWNs to the zone.
7. Repeat steps 4-6 for all remaining peer zones.
8. Click the Zone Config tab.
9. Click New Zone Config, and enter a zone configuration name.
10. Select all newly created zones.
11. Click the right arrow to add the peer zones to the zone config.
12. Click Zoning Actions→Save Config to save the changes.
13. Click Zoning Actions→Enable Config.
14. Click Yes.

Configuring the storage

Once we established all FC connectivity, we configured each array for our test environment. For the VMAX configuration, we deployed a standalone Unisphere appliance deployed on our infrastructure server to manage our storage as well as act as our witness for the SRDF/Metro testing. Similarly, we configured a standalone 3PAR StoreServ Management Console VM to manage the 3PAR arrays. We configured each array as RAID5(3+1) to match across both environments and added our host information for mapping/exporting our volumes.

We configured the following volumes for each VM with thin provisioning and compression enabled:

VM configuration information	
Oracle VM 1	
OS	85 GB
Data1-Data16	100 GB
Log1-Log8	50 GB
FRA1-FRA4	150 GB
Oracle VM 2	
OS	85 GB
Data1-Data16	100 GB
Log1-Log8	50 GB
FRA1-FRA4	150 GB
Data mart VM	
OS	85 GB
Data1-Data8	500 GB
Log	300 GB

About our Oracle 12c database configuration

We created two new VMs, installed Oracle Enterprise Linux® 7.3 on each VM, and configured each VM with 64GB RAM and 24 vCPUs. Each VM had an 80GB virtual disk, upon which we installed the Oracle OS and database software. This is the only VMDK we created for each VM. We deployed it on the OS datastore along with the VM configuration files. We then added sixteen 100GB RDMs to hold the database data files, eight 50GB RDMs to store the database log files, and four 150GB RDMs for the fast recovery area to store the archive logs.

We made the necessary networking adjustments and prepared the VM for Oracle installation prerequisites. In addition, we configured other OS settings, such as stopping unneeded services, setting the system's kernel parameters in sysctl.conf, and reserving huge pages for Oracle.

We installed Oracle Grid Infrastructure 12c for Linux x86-64 on the VM and created Oracle ASM disk groups for data, logs, and the fast recovery area. We then installed and configured the Oracle database. For each database instance, we set the database to use large pages only. For the detailed spfile, see Appendix E.

Finally, we used the SLOB 2.3 tool to generate the initial database schema. We used a benchmark scale of 9,600MB in SLOB's workload generator to create approximately 1.2TB of data, and we configured SLOB to create the tables and indices. For details, including the SLOB configuration file, see Appendix D and Appendix F.

About the SLOB 2.3 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 https://www.kevinclosson.net/slob/.

Appendix D: Detailed test procedure

We used the following steps to configure each server and our Oracle test environment.

Configuring the servers

Installing VMware vSphere 6.5

We installed the VMware vSphere 6.5 hypervisor to a local RAID1 disk pair. We created the RAID1 virtual disk using the BIOS utilities on each server.

1. Boot the server to the installation media.
2. At the boot menu screen, choose the standard installer.
3. Press Enter.
4. Press F11 to accept the license terms.
5. Press Enter to install to the local virtual disk.
6. Press Enter to select the US Default keyboard layout.
7. Create a password for the root user and press Enter.
8. Press F11 to begin the installation.
9. Once installation is complete, login to the management console, configure the management network, and enable both SSH and esxcli.

Configuring multi-pathing

We configured each server to use multi-pathing for each storage array using best practices. For the Dell EMC servers, we installed Dell EMC PowerPath/VE to handle multi-pathing to the VMAX 250F. For the HPE servers, we created a custom storage policy using the following steps in ESXCLI:

1. Log into each server using SSH.
2. Run the following command:

   ```
   esxcli storage nmp satp rule add -s "VMW_SATP_ALUA" -P "VMW_PSP_RR" -O "iops=1" -c "tpgs_on" -V
   "3PARdata" -M "VV" -e "HPE 3PAR Custom Rule"
   ```
3. Run the following command to verify that you have created the custom rule:

   ```
   esxcli storage nmp satp rule list | grep "3PARdata"
   ```

Installing VMware vCenter® 6.5

We used the following steps to deploy the VMware vCenter appliance on our infrastructure server. We deployed one for the Dell EMC environment and one for the HPE environment.

1. Mount the VCSA ISO to a Windows server that has connectivity to the target vCenter host.
2. Browse to <mount location>\vcsa-ui-installer\win32 and run `installer.exe`.
3. Click Install.
4. Click Next.
5. Check I accept the terms of the license agreement, and click Next.
6. Leave vCenter Server with an Embedded Platform Services Controller checked, and click Next.
7. Enter the IP address and credentials for the target vCenter host, and click Next.
8. Enter a name and password for the VCSA appliance, and click Next.
9. Select Tiny for the deployment size, and click Next.
10. Select a datastore to store the appliance, and click Next.
11. Enter network settings for the appliance, and click Next.
12. Review the summary and click Finish.
13. Once the deployment is complete, click Continue.
14. Click Next.
15. Select Synchronize time with NTP servers from the drop-down menu and enter the IP address or hostnames of your NTP servers. Select Enabled from the SSH drop-down menu. Click Next.
16. Enter a username and password for the vCenter SSO, and click Next.
17. Uncheck Join the VMware's Customer Experience Improvement Program (CEIP), and click Next.
18. Once each vCenter completed the deployment process, add the hosts to vCenter to manage them.
19. Rescan the host storage, create a datastore for the OS volumes, and configure the rest of the volumes as RDMs on the VMs.

Creating the Oracle VMs

We created one gold image VM for Oracle using the following steps:

1. In VMware vCenter, navigate to Virtual Machines.
2. To create a new VM, click the icon.
3. Leave Create a new virtual machine selected, and click Next.
4. Enter a name for the virtual machine, and click Next.
5. Place the VM on the desired host with available CPUs, and click Next.
6. Select the Dell EMC VMAX 250F LUN for the 80GB OS VMDK, and click next.
7. Select the guest OS family as Linux and Guest OS Version as Oracle Linux 7 (64-bit), and click Next.
8. In the Customize Hardware section, use the following settings:
   a. Set the vCPU count to 24.
   b. Set the Memory to 64GB, and check the Reserve all guest memory box.
   c. Add 16 x 100GB RDMs, 8 x 50GB RDMs, and 4 x 150GB RDMs.
   d. Add three additional VMware Paravirtual SCSI conrollers and split the 100GB RDMs between two, all eight 50GB RDMs to the third controller, and the four 150GB RDMs to the first original controller.
   e. Attach an Oracle Linux 7.3 ISO to the CD/DVD drive.
9. Click Next.
10. Click Finish.
11. Power on the VMs, and follow the steps outlined in the next section to install and configure the workload.

Installing Oracle Enterprise Linux 7.3

1. Attach the Oracle Enterprise Linux 7.3 ISO to the VM, and boot to it.
2. Select Install or upgrade an existing system.
3. Choose the language you wish to use, and click Continue.
4. Select Date & Time, and ensure the VM is set to the correct date, time, and time zone.
5. Click Done.
6. Select Installation Destination.
7. Select the desired virtual disk for the OS.
8. Under Other Storage Options, select I will configure partitioning.
9. Click Done.
10. Select Click here to create them automatically.
11. Remove the /home partition.
12. Expand the swap partition to 20GB.
13. Assign all remaining free space to the / partition.
14. Click Done.
15. Click Accept Changes.
16. Select Kdump.
17. Uncheck Enable kdump, and click Done.
18. Select Network & Hostname.
19. Enter the desired hostname for the VM.
20. Turn on the desired network port, and click Configure.
21. On the General tab, select Automatically connect to this network when it is available.
22. On the IPv4 Settings tab, select Manual under Method.
23. Under Addresses, click Add, and enter the desired static IP information for the VM.
24. Enter the desired DNS information.
25. Click save, and click Done.
26. Click Begin Installation.
27. Select Root Password.
28. Enter the desired root password, and click Done.
29. When the installation completes, select Reboot to restart the server.

Configuring OEL 7.3 for Oracle

1. Log onto the server as root.
2. Disable the firewall:

```
systemctl stop firewalld
systemctl disable firewalld
```

3. Disable SELinux:

```
vi /etc/selinux/config
SELINUX=disabled
```

4. Update OEL 7.3:

```
yum update
```

5. Install VMware tools:

```
yum install open-vm-tools
```

6. Install the Oracle 12c preinstall RPM:

```
yum install oracle-database-server-12cR2-preinstall
```

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
```

8. Disable auditd:

```
systemctl disable auditd
```

9. Create Oracle users and groups by running these shell commands:

```
groupadd -g 54323 oper
groupadd -g 54327 asmdba
groupadd -g 54328 asmoper
groupadd -g 54329 asmadmin
usermod -u 54321 -g oinstall -G dba,oper,asmdba,asmoper,asmadmin oracle
```

10. Add the following lines to the .bash_profile file for the oracle user:

```
export TMP=/tmp
export TMPDIR=$TMP

export ORACLE_HOSTNAME=oracle.koons.local
export ORACLE_UNQNAME=orcl
export ORACLE_BASE=/u01/app/oracle
export GRID_HOME=/u01/app/12.2.0.2/grid
export DB_HOME=$ORACLE_BASE/product/12.2.0.2/dbhome_1
export ORACLE_HOME=$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'
```

11. Create the following files in the oracle user's home folder.

```
>>>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
>>>grid_env<<<
export ORACLE_SID=+ASM
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
```

12. Create the following directories, and assign the following permissions.

```
mkdir -p  /u01/app/12.1.0.2/grid
mkdir -p /u01/app/oracle/product/12.1.0.2/dbhome_1
chown -R oracle:oinstall /u01
chmod -R 775 /u01/
```

13. Create passwords for the oracle account with `passwd`.
14. Append the following to `/etc/security/limits.conf`:

```
oracle -  nofile  65536
oracle -  nproc   16384
oracle -  stack   32768
oracle -  memlock 152043520
* soft memlock unlimited
* hard memlock unlimited
```

15. Modify the system's kernel parameters by appending the following to `/etc/sysctl.conf`:

```
vm.nr_hugepages = 14336
vm.hugetlb_shm_group = 54321
```

16. Install the `oracleasmlib` packages:

```
yum install -y oracleasm-support-* oracleasmlib-*
```

17. Create a partition on all VMDKs using fdisk.
18. Edit `/etc/sysconfig/oracleasm` to contain the following:

```
# ORACLEASM_ENABLED: 'true' means to load the driver on boot.
ORACLEASM_ENABLED=true

# ORACLEASM_UID: Default UID owning the /dev/oracleasm mount point.
ORACLEASM_UID=oracle

# ORACLEASM_GID: Default GID owning the /dev/oracleasm mount point.
ORACLEASM_GID=oinstall


# ORACLEASM_SCANBOOT: 'true' means fix disk perms on boot
ORACLEASM_SCANBOOT=true

# ORACLEASM_USE_LOGICAL_BLOCK_SIZE: 'true' means use the logical block
# size reported by the underlying disk instead of the physical. The
# default is 'false' ORACLEASM_USE_LOGICAL_BLOCK_SIZE=false
```

19. Run the following commands to configure all disks for Oracle ASM:

```
oracleasm createdisk FRA1 /dev/sdb1
oracleasm createdisk FRA2 /dev/sdc1
oracleasm createdisk FRA3 /dev/sdd1
oracleasm createdisk FRA4 /dev/sde1
oracleasm createdisk DATA1 /dev/sdf1
oracleasm createdisk DATA2 /dev/sdg1
oracleasm createdisk DATA3 /dev/sdh1
oracleasm createdisk DATA4 /dev/sdi1
oracleasm createdisk DATA5 /dev/sdj1
oracleasm createdisk DATA6 /dev/sdk1
oracleasm createdisk DATA7 /dev/sdl1
oracleasm createdisk DATA8 /dev/sdm1
oracleasm createdisk DATA9 /dev/sdn1
oracleasm createdisk DATA10 /dev/sdo1
oracleasm createdisk DATA11 /dev/sdp1
oracleasm createdisk DATA12 /dev/sdq1
oracleasm createdisk DATA13 /dev/sdr1
oracleasm createdisk DATA14 /dev/sds1
oracleasm createdisk DATA15 /dev/sdt1
oracleasm createdisk DATA16 /dev/sdu1
oracleasm createdisk LOG1 /dev/sdv1
oracleasm createdisk LOG2 /dev/sdw1
oracleasm createdisk LOG3 /dev/sdx1
oracleasm createdisk LOG4 /dev/sdy1
oracleasm createdisk LOG5 /dev/sdz1
oracleasm createdisk LOG6 /dev/sdaa1
oracleasm createdisk LOG7 /dev/sdab1
oracleasm createdisk LOG8 /dev/sdac1
```

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. In the Select Installation Option screen, select Install and Configure Grid Infrastructure for a Standalone Server, and click Next.
7. Choose the language, and click Next.
8. In the Create ASM Disk Group screen, choose the Disk Group Name, and change redundancy to External.
9. Change the path to `/dev/oracleasm/disks`, select the sixteen disks that you are planning to use for the database, and click Next.
10. In the Specify ASM Password screen, choose Use same password for these accounts, write the passwords for the ASM users, and click Next.
11. At the Management Options screen, 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. At the Prerequisite Checks screen, make sure that there are no errors.
17. At 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
19. run the scripts.
20. At the Finish screen, click Close.
21. To run the ASM Configuration Assistant, type `asmca`.
22. In the ASM Configuration Assistant, click Create.
23. In the Create Disk Group window, name the new disk group `log`, choose redundancy External (None), select the eight disks for redo logs.
24. In the ASM Configuration Assistant, click Create.
25. In the Create Disk Group window, name the new disk group `FRA`, choose redundancy External (None), select the four disks for the fast recovery area.
26. 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 to load.
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. At the Download Software Updates screen, enter the desired update option, and click Next.
9. At the Select Installation Option screen, select Install database software only, and click Next.
10. At the Grid Installation Options screen, select Single instance database installation, and click Next.
11. At the Select Product Languages screen, leave the default setting of English, and click Next.
12. At the Select Database Edition screen, select Enterprise Edition, and click Next.
13. At the Specify Installation Location, leave the defaults, and click Next.
14. At the Create Inventory screen, leave the default settings, and click Next.
15. At the Privileged Operating System groups screen, keep the defaults, and click Next.
16. Allow the prerequisite checker to complete.
17. At the Summary screen, click Install.
18. Once the Execute Configuration scripts prompt appears, ssh into the server as `root`, and run the following command:

    `# /u01/app/oracle/product/12.2.0.2/dbhome_1/root.sh`

19. Return to the prompt, and click OK.
20. Once the installer completes, click Close.

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 Deployment Type screen, select General Purpose or Transaction Processing. Click Next.
6. Enter a Global database name and the appropriate SID, and uncheck Create as Container database. Click Next.
7. At the storage option screen, select Use following for the database storage attributes.
8. In the drop down, select Automatic Storage Management (ASM), and select +DATA for the file location.
9. At the Fast Recovery Option screen, check the box for Specify Fast Recovery Area.
10. In the drop down, select ASM, select +FRA for the Fast Recovery Area, and enter 590 GB for the size.
11. Check the box for Enable Archiving, and click Next.
12. At the Network Configuration screen, select the listener, and click Next.
13. At the Data Vault Option screen, leave as default, and click Next.
14. At the Configuration Options screen, leave the default memory settings, and click Next.
15. At the Management Options screen select Configure Enterprise Manager (EM) Database Express, and click Next.
16. At the User Credentials screen, select Use the same administrative password for all accounts, enter and confirm the desired password, and click Next.
17. At the Creation Options, select Create Database, and click Next.
18. At the summary screen click Finish.
19. Close the Database Configuration Assistant.
20. In a Web browser, browse to `https://vm.ip.address:5500/em` to open the database manager.
21. Log in as `sys` with the password you specified.
22. Go to Storage→Tablespaces.
23. Click Create.
24. Enter SLOB as the Name of the tablespace, and check the Set As Default box.
25. Add 40 Oracle-managed files sized at 32767M. Click OK.
26. Go to Storage→Redo Log Groups.
27. Click Actions→Switch file… until you get one of the groups to go inactive.
28. Highlight the inactive group, and click Actions→Drop group.
29. Create four redo log groups, each with a single 10GB file on the +LOG ASM volume.
30. Repeat steps 23 and 24, removing the remaining default redo logs.

Installing SLOB and populating the database

1. Download the SLOB kit from http://kevinclosson.net/slob/
2. Copy and untar the files to /home/oracle/SLOB.
3. Edit the slob.conf file to match Appendix F.
4. Type `./setup.sh SLOB 24` to start the data population to the SLOB tablespace we created earlier.
5. The database is populated when the setup is complete.

Configuring the bulk load data mart VM

We created a separate data mart load VM on the third server in each environment. We doubled the RAM in each data mart host server to 512GB. Additionally, we installed three 2TB PCIe SSDs to hold the 1.8TB VMDKs for the HammerDB data generation. We configured each VM with 64 vCPUs and 480 GB of RAM (reserved). Each VM had a 60GB VMDK for OS, eight 500GB RDMs for data, and one 300GB RDM for logs.

We used HammerDB to generate TPC-H-compliant source data at scale factor 3,000, for a total of 3.31 TB of raw data. The generated data exists in the form of pipe-delimited text files, which we placed on NVMe PCIe SSDs for fast reads. We split the six largest tables into 64 separate files for parallel loading. Each chunk had its own table in SQL Server, for a total of 6 x 64 one-to-one streams. We used batch scripting and SQLCMD to start 64 simultaneous SQL scripts. Each script contained BULK INSERT statements to load the corresponding chunk for each table. For example, the 17th SQL script loaded ORDERS_17.txt into table ORDERS_17, and upon finishing, began loading LINEITEM_17.txt into table LINEITEM_17, and so on through each table.

Installing Microsoft® Windows Server® 2016 Datacenter Edition

1. Boot the VM to the installation media.
2. Press any key when prompted to boot from DVD.
3. When the installation screen appears, leave language, time/currency format, and input method as default, and click Next.

4. Click Install now.
5. When the installation prompts you, enter the product key.
6. Check I accept the license terms, and click Next.
7. Click Custom: Install Windows only (advanced).
8. Select Windows Server 2016 Datacenter Edition (Desktop Experience), and click Next.
9. Select Drive 0 Unallocated Space, and click Next. Windows will start and restart automatically after completion.
10. When the Settings page appears, fill in the Password and Reenter Password fields with the same password.
11. Log in with the password you set up previously.

Installing Microsoft SQL Server® 2016

1. Prior to installing, add the .NET Framework 3.5 feature to the server.
2. Mount the installation DVD for SQL Server 2016.
3. Click Run SETUP.EXE. If Autoplay does not begin the installation, navigate to the SQL Server 2016 DVD, and double-click it.
4. In the left pane, click Installation.
5. Click New SQL Server stand-alone installation or add features to an existing installation.
6. Select the Enter the product key radio button, enter the product key, and click Next.
7. Click the checkbox to accept the license terms, and click Next.
8. Click Use Microsoft Update to check for updates, and click Next.
9. Click Install to install the setup support files.
10. If no failures show up, click Next.
11. At the Setup Role screen, choose SQL Server Feature Installation, and click Next.
12. At the Feature Selection screen, select Database Engine Services, Full-Text and Semantic Extractions for Search, Client Tools Connectivity, Client Tools Backwards Compatibility. Click Next.
13. At the Installation Rules screen, after the check completes, click Next.
14. At the Instance configuration screen, leave the default selection of default instance, and click Next.
15. At the Server Configuration screen, choose NT Service\SQLSERVERAGENT for SQL Server Agent, and choose NT Service\MSSQLSERVER for SQL Server Database Engine. Change the Startup Type to Automatic. Click Next.
16. At the Database Engine Configuration screen, select the authentication method you prefer. For our testing purposes, we selected Mixed Mode.
17. Enter and confirm a password for the system administrator account.
18. Click Add Current user. This may take several seconds.
19. Click Next.
20. At the Error and usage reporting screen, click Next.
21. At the Installation Configuration Rules screen, check that there are no failures or relevant warnings, and click Next.
22. At the Ready to Install screen, click Install.
23. After installation completes, click Close.
24. Close the installation window.

Generating the data

1. Download HammerDB v2.21 and run `hammerdb.bat`.
2. Click Options→Benchmark.
3. Select the radio buttons for MSSQL Server and TPC-H, and click OK.
4. In the left pane, expand SQL Server→TPC-H→Datagen, and double-click Options.
5. Select the radio button for 3000, enter a target location for the TPC-H data, and select 64 for the number of virtual users. Click OK.
6. Double-click Generate and click Yes.

Creating the target database

We used the following SQL script to create the target database (some lines have been removed and replaced with an ellipsis for clarity):

```
IF EXISTS (SELECT name FROM master.dbo.sysdatabases WHERE name = 'tpch3000')
    DROP DATABASE tpch3000
GO

CREATE DATABASE tpch3000
ON PRIMARY
(   NAME       = tpch3000_root,
    FILENAME   = 'F:\tpch\tpch_root.mdf',
    SIZE       = 100MB,
    FILEGROWTH = 100MB),
```

```
FILEGROUP    DATA_FG_MISC
(   NAME          = tpch3000_data_ms,
    FILENAME    = 'F:\tpch\tpch_data_ms.mdf',
    SIZE        = 500MB,
    FILEGROWTH  = 100MB),

FILEGROUP DATA_FG_01 (NAME = tpch3000_data_01, FILENAME = 'F:\tpch\tpch_data_01.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_02 (NAME = tpch3000_data_02, FILENAME = 'G:\tpch\tpch_data_02.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_03 (NAME = tpch3000_data_03, FILENAME = 'H:\tpch\tpch_data_03.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_04 (NAME = tpch3000_data_04, FILENAME = 'I:\tpch\tpch_data_04.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_05 (NAME = tpch3000_data_05, FILENAME = 'J:\tpch\tpch_data_05.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_06 (NAME = tpch3000_data_06, FILENAME = 'K:\tpch\tpch_data_06.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_07 (NAME = tpch3000_data_07, FILENAME = 'L:\tpch\tpch_data_07.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_08 (NAME = tpch3000_data_08, FILENAME = 'M:\tpch\tpch_data_08.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),

...

'F:\tpch\tpch_data_57.mdf', SIZE = 55296MB, FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_58 (NAME = tpch3000_data_58, FILENAME = 'G:\tpch\tpch_data_58.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_59 (NAME = tpch3000_data_59, FILENAME = 'H:\tpch\tpch_data_59.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_60 (NAME = tpch3000_data_60, FILENAME = a'I:\tpch\tpch_data_60.mdf', SIZE =
55296MB, FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_61 (NAME = tpch3000_data_61, FILENAME = 'J:\tpch\tpch_data_61.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_62 (NAME = tpch3000_data_62, FILENAME = 'K:\tpch\tpch_data_62.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_63 (NAME = tpch3000_data_63, FILENAME = 'L:\tpch\tpch_data_63.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB),
FILEGROUP DATA_FG_64 (NAME = tpch3000_data_64, FILENAME = 'M:\tpch\tpch_data_64.mdf', SIZE = 55296MB,
FILEGROWTH  = 100MB)
LOG ON
(   NAME        = tpch3000_log,
    FILENAME    = 'N:\LOG\tpch3000\tpch3000_log.ldf',
    SIZE        = 290GB,
    FILEGROWTH  = 100MB)

GO

/*set db options*/
ALTER DATABASE tpch3000 SET RECOVERY SIMPLE
ALTER DATABASE tpch3000 SET AUTO_CREATE_STATISTICS OFF
ALTER DATABASE tpch3000 SET AUTO_UPDATE_STATISTICS OFF
ALTER DATABASE tpch3000 SET PAGE_VERIFY NONE

USE tpch3000
GO

create table CUSTOMER_1 ([c_custkey] [bigint] NOT NULL,[c_mktsegment] [char](10) NULL,[c_nationkey]
[int] NULL,[c_name] [varchar](25) NULL,[c_address] [varchar](40) NULL,[c_phone] [char](15) NULL,[c_
acctbal] [money] NULL,[c_comment] [varchar](118) NULL) on DATA_FG_01

create table CUSTOMER_2 ([c_custkey] [bigint] NOT NULL,[c_mktsegment] [char](10) NULL,[c_nationkey]
[int] NULL,[c_name] [varchar](25) NULL,[c_address] [varchar](40) NULL,[c_phone] [char](15) NULL,[c_
acctbal] [money] NULL,[c_comment] [varchar](118) NULL) on DATA_FG_02

...
```

```
create table CUSTOMER_64 ([c_custkey] [bigint] NOT NULL,[c_mktsegment] [char](10) NULL,[c_nationkey]
[int] NULL,[c_name] [varchar](25) NULL,[c_address] [varchar](40) NULL,[c_phone] [char](15) NULL,[c_
acctbal] [money] NULL,[c_comment] [varchar](118) NULL) on DATA_FG_64

create table LINEITEM_1 ([l_shipdate] [date] NULL,[l_orderkey] [bigint] NOT NULL,[l_discount] [money]
NOT NULL,[l_extendedprice] [money] NOT NULL,[l_suppkey] [int] NOT NULL,[l_quantity] [bigint] NOT
NULL,[l_returnflag] [char](1) NULL,[l_partkey] [bigint] NOT NULL,[l_linestatus] [char](1) NULL,[l_
tax] [money] NOT NULL,[l_commitdate] [date] NULL,[l_receiptdate] [date] NULL,[l_shipmode] [char]
(10) NULL,[l_linenumber] [bigint] NOT NULL,[l_shipinstruct] [char](25) NULL,[l_comment] [varchar](44)
NULL) on DATA_FG_01
create table LINEITEM_2 ([l_shipdate] [date] NULL,[l_orderkey] [bigint] NOT NULL,[l_discount] [money]
NOT NULL,[l_extendedprice] [money] NOT NULL,[l_suppkey] [int] NOT NULL,[l_quantity] [bigint] NOT
NULL,[l_returnflag] [char](1) NULL,[l_partkey] [bigint] NOT NULL,[l_linestatus] [char](1) NULL,[l_
tax] [money] NOT NULL,[l_commitdate] [date] NULL,[l_receiptdate] [date] NULL,[l_shipmode] [char]
(10) NULL,[l_linenumber] [bigint] NOT NULL,[l_shipinstruct] [char](25) NULL,[l_comment] [varchar](44)
NULL) on DATA_FG_02

...

create table LINEITEM_64 ([l_shipdate] [date] NULL,[l_orderkey] [bigint] NOT NULL,[l_discount]
[money] NOT NULL,[l_extendedprice] [money] NOT NULL,[l_suppkey] [int] NOT NULL,[l_quantity] [bigint]
NOT NULL,[l_returnflag] [char](1) NULL,[l_partkey] [bigint] NOT NULL,[l_linestatus] [char](1) NULL,[l_
tax] [money] NOT NULL,[l_commitdate] [date] NULL,[l_receiptdate] [date] NULL,[l_shipmode] [char]
(10) NULL,[l_linenumber] [bigint] NOT NULL,[l_shipinstruct] [char](25) NULL,[l_comment] [varchar](44)
NULL) on DATA_FG_64

create table ORDERS_1 ([o_orderdate] [date] NULL,[o_orderkey] [bigint] NOT NULL,[o_custkey] [bigint]
NOT NULL,[o_orderpriority] [char](15) NULL,[o_shippriority] [int] NULL,[o_clerk] [char](15) NULL,[o_
orderstatus] [char](1) NULL,[o_totalprice] [money] NULL,[o_comment] [varchar](79) NULL) on DATA_FG_01
create table ORDERS_2 ([o_orderdate] [date] NULL,[o_orderkey] [bigint] NOT NULL,[o_custkey] [bigint]
NOT NULL,[o_orderpriority] [char](15) NULL,[o_shippriority] [int] NULL,[o_clerk] [char](15) NULL,[o_
orderstatus] [char](1) NULL,[o_totalprice] [money] NULL,[o_comment] [varchar](79) NULL) on DATA_FG_02

...

create table ORDERS_64 ([o_orderdate] [date] NULL,[o_orderkey] [bigint] NOT NULL,[o_custkey] [bigint]
NOT NULL,[o_orderpriority] [char](15) NULL,[o_shippriority] [int] NULL,[o_clerk] [char](15) NULL,[o_
orderstatus] [char](1) NULL,[o_totalprice] [money] NULL,[o_comment] [varchar](79) NULL) on DATA_FG_64

create table PART_1 ([p_partkey] [bigint] NOT NULL,[p_type] [varchar](25) NULL,[p_size] [int]
NULL,[p_brand] [char](10) NULL,[p_name] [varchar](55) NULL,[p_container] [char](10) NULL,[p_mfgr]
[char](25) NULL,[p_retailprice] [money] NULL,[p_comment] [varchar](23) NULL) on DATA_FG_01
create table PART_2 ([p_partkey] [bigint] NOT NULL,[p_type] [varchar](25) NULL,[p_size] [int]
NULL,[p_brand] [char](10) NULL,[p_name] [varchar](55) NULL,[p_container] [char](10) NULL,[p_mfgr]
[char](25) NULL,[p_retailprice] [money] NULL,[p_comment] [varchar](23) NULL) on DATA_FG_02

...

create table PART_64 ([p_partkey] [bigint] NOT NULL,[p_type] [varchar](25) NULL,[p_size] [int]
NULL,[p_brand] [char](10) NULL,[p_name] [varchar](55) NULL,[p_container] [char](10) NULL,[p_mfgr]
[char](25) NULL,[p_retailprice] [money] NULL,[p_comment] [varchar](23) NULL) on DATA_FG_64

create table PARTSUPP_1 ([ps_partkey] [bigint] NOT NULL,[ps_suppkey] [int] NOT NULL,[ps_supplycost]
[money] NOT NULL,[ps_availqty] [int] NULL,[ps_comment] [varchar](199) NULL) on DATA_FG_01
create table PARTSUPP_2 ([ps_partkey] [bigint] NOT NULL,[ps_suppkey] [int] NOT NULL,[ps_supplycost]
[money] NOT NULL,[ps_availqty] [int] NULL,[ps_comment] [varchar](199) NULL) on DATA_FG_02

...

create table PARTSUPP_64 ([ps_partkey] [bigint] NOT NULL,[ps_suppkey] [int] NOT NULL,[ps_supplycost]
[money] NOT NULL,[ps_availqty] [int] NULL,[ps_comment] [varchar](199) NULL) on DATA_FG_64

create table SUPPLIER_1 ([s_suppkey] [int] NOT NULL,[s_nationkey] [int] NULL,[s_comment] [varchar]
(102) NULL,[s_name] [char](25) NULL,[s_address] [varchar](40) NULL,[s_phone] [char](15) NULL,[s_
acctbal] [money] NULL) on DATA_FG_01
```

```
create table SUPPLIER_2 ([s_suppkey] [int] NOT NULL,[s_nationkey] [int] NULL,[s_comment] [varchar]
(102) NULL,[s_name] [char](25) NULL,[s_address] [varchar](40) NULL,[s_phone] [char](15) NULL,[s_
acctbal] [money] NULL) on DATA_FG_02

...

create table SUPPLIER_64 ([s_suppkey] [int] NOT NULL,[s_nationkey] [int] NULL,[s_comment] [varchar]
(102) NULL,[s_name] [char](25) NULL,[s_address] [varchar](40) NULL,[s_phone] [char](15) NULL,[s_
acctbal] [money] NULL) on DATA_FG_64
```

Inserting the data into Microsoft SQL Server
We used 64 individual SQL scripts to create a BULK INSERT process on each filegroup. The first script shown here is an example:

```
bulk insert tpch3000..CUSTOMER_1 from 'O:\CUSTOMER_1.tbl' with
(TABLOCK,DATAFILETYPE='char',CODEPAGE='raw',FieldTerminator='|',BATCHSIZE=14062500)
bulk insert tpch3000..LINEITEM_1 from 'O:\LINEITEM_1.tbl' with
(TABLOCK,DATAFILETYPE='char',CODEPAGE='raw',FieldTerminator='|',BATCHSIZE=562500000)
bulk insert tpch3000..ORDERS_1 from 'O:\ORDERS_1.tbl' with
(TABLOCK,DATAFILETYPE='char',CODEPAGE='raw',FieldTerminator='|',BATCHSIZE=140625000)
bulk insert tpch3000..PART_1 from 'O:\PART_1.tbl' with
(TABLOCK,DATAFILETYPE='char',CODEPAGE='raw',FieldTerminator='|',BATCHSIZE=18750000)
bulk insert tpch3000..PARTSUPP_1 from 'O:\PARTSUPP_1.tbl' with
(TABLOCK,DATAFILETYPE='char',CODEPAGE='raw',FieldTerminator='|',BATCHSIZE=75000000)
bulk insert tpch3000..SUPPLIER_1 from 'O:\SUPPLIER_1.tbl' with
(TABLOCK,DATAFILETYPE='char',CODEPAGE='raw',FieldTerminator='|',BATCHSIZE=937500)
```

Starting the SQL BULK INSERT scripts
We used Windows CMD and SQLCMD to start the 64 BULK INSERT scripts with CPU affinity:

```
start /node 0 /affinity 1 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_1.sql
start /node 0 /affinity 2 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_2.sql
start /node 0 /affinity 4 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_3.sql
start /node 0 /affinity 8 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_4.sql
start /node 0 /affinity 10 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_5.sql
start /node 0 /affinity 20 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_6.sql
start /node 0 /affinity 40 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_7.sql
start /node 0 /affinity 80 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_8.sql
start /node 0 /affinity 100 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_9.sql
start /node 0 /affinity 200 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_10.sql
start /node 0 /affinity 400 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_11.sql
start /node 0 /affinity 800 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_12.sql
start /node 0 /affinity 1000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_13.sql
start /node 0 /affinity 2000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_14.sql
start /node 0 /affinity 4000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_15.sql
start /node 0 /affinity 8000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_16.sql
start /node 1 /affinity 1 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_17.sql
start /node 1 /affinity 2 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_18.sql
start /node 1 /affinity 4 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_19.sql
start /node 1 /affinity 8 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_20.sql
```

```
start /node 1 /affinity 10 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_21.sql
start /node 1 /affinity 20 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_22.sql
start /node 1 /affinity 40 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_23.sql
start /node 1 /affinity 80 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_24.sql
start /node 1 /affinity 100 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_25.sql
start /node 1 /affinity 200 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_26.sql
start /node 1 /affinity 400 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_27.sql
start /node 1 /affinity 800 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_28.sql
start /node 1 /affinity 1000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_29.sql
start /node 1 /affinity 2000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_30.sql
start /node 1 /affinity 4000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_31.sql
start /node 1 /affinity 8000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_32.sql
start /node 2 /affinity 1 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_33.sql
start /node 2 /affinity 2 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_34.sql
start /node 2 /affinity 4 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_35.sql
start /node 2 /affinity 8 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_36.sql
start /node 2 /affinity 10 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_37.sql
start /node 2 /affinity 20 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_38.sql
start /node 2 /affinity 40 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_39.sql
start /node 2 /affinity 80 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_40.sql
start /node 2 /affinity 100 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_41.sql
start /node 2 /affinity 200 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_42.sql
start /node 2 /affinity 400 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_43.sql
start /node 2 /affinity 800 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_44.sql
start /node 2 /affinity 1000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_45.sql
start /node 2 /affinity 2000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_46.sql
start /node 2 /affinity 4000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_47.sql
start /node 2 /affinity 8000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_48.sql
start /node 3 /affinity 1 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_49.sql
start /node 3 /affinity 2 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_50.sql
start /node 3 /affinity 4 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_51.sql
start /node 3 /affinity 8 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_52.sql
start /node 3 /affinity 10 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_53.sql
start /node 3 /affinity 20 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
```

```
Administrator\Documents\gen_54.sql
start /node 3 /affinity 40 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_55.sql
start /node 3 /affinity 80 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_56.sql
start /node 3 /affinity 100 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_57.sql
start /node 3 /affinity 200 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_58.sql
start /node 3 /affinity 400 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_59.sql
start /node 3 /affinity 800 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_60.sql
start /node 3 /affinity 1000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_61.sql
start /node 3 /affinity 2000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_62.sql
start /node 3 /affinity 4000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_63.sql
start /node 3 /affinity 8000 sqlcmd -S localhost -d tpch3000 -U sa -P Password1 -i C:\Users\
Administrator\Documents\gen_64.sql
```

Running the SLOB and data mart testing

1. Reboot all test VMs and hosts before each test.
2. After rebooting everything, power up the Oracle and data mart VMs.
3. Wait 10 minutes.
4. Start esxtop on all hosts with the following script adjusting the number of iterations to match the test run time:

   ```
   esxtop -a -b -n 185 -d 60 > esxout.`date +"%Y-%m-%d_%H-%M-%S"`
   ```

5. Additionally, we configured a user defined data collector set in Windows on the data mart VM to gather disk and CPU metrics at the VM level. Start the collector set with a stop condition of 3 hours.
6. Once all performance-gathering tools have started, log into each Oracle VM as the `oracle` user, and navigate to `/home/oracle/SLOB`.
7. Run `./runit.sh 24` on each VM.
8. Allow SLOB to run for 30 minutes.
9. After 30 minutes, run the start_bulk_insters.bat script on the data mart VM.
10. The data mart load will then start 64 simultaneous streams to the database with large block writes.
11. The load will finish in about 1 hour and 45 minutes.
12. Allow the SLOB runs and host performance gathering tools to complete.
13. Copy output data from the VMs and the hosts.
14. Use Unishpere to grab historical performance data from the VMAX array, and use SSMC to grab historical data from the 3PAR array.
15. Using the SQL query "truncate table" on each SQL table to clean up the data mart database in between runs.
16. As the `oracle` user on the Oracle VMs, use rman to delete the archive logs in between test runs.
17. We reported IOPS from the combined iostat output from the SLOB VMs. We reported latency from the esxtop data for the storage FC adapters.

Configuring the VMAX SRDF/Metro and 3PAR Peer Persistence testing

Creating the SRDF/Metro connection

1. Log into the Unisphere for VMAX of the site A storage array.
2. Select All Symmetrix→your site A storage array.
3. Select Data Protection→Create SRDF Group
4. In the Create SRDF Group pop-up, choose the following options:

   • Communication Protocol: FC
   • SRDF Group Label: Your choice
   • SRDF Group Number: Your choice (make sure to choose a number not already chosen, though)
   • Director: Choose all available RDF directors
   • Remote Symmetrix ID: Your site B storage array
   • Remote SRDF Group Number: the same number as SRDF Group Number
   • Remote Director: Choose all available remote RDF directors

5. Click OK.
6. Select Data Protection→Protection Dashboard.
7. In the Protection Dashboard, click on Unprotected.
8. Select the storage group you are trying to protect, and click Protect.
9. In the Protect Storage Group window, select High Availability Using SRDF/Metro, and click Next.
10. In the Select SRDF/Metro Connectivity step, select your site B storage, make sure to protect your SRDF group via Witness, enable compression, and click Next.
11. In the Finish step, click the dropdown list beside Add to Job List, and select Run Now.
12. Select Data Protection→SRDF
13. Select SRDF/Metro.
14. Monitor the state of the SRDF metro pair you made. Proceed to the next step once the pair reads as ActiveActive.
15. Select All Symmetrix→your site B storage array.
16. Select Storage→Storage Groups Dashboard
17. Double-click on the newly-synced SRDF storage group.
18. Select Provision Storage to Host.
19. In Select Host/Host Group, select your site A server.
20. In Select Port Group, allow the automatic creation of a new port group.
21. In Review, click the down arrow beside Add to Job List, and click Run Now.

Creating the 3PAR Peer Persistence connection

Prior to completing these steps, we configured a virtual witness by deploying the 3PAR Remote Copy Quorum Witness appliance on our infrastructure server. We ensured that the appliance had network connectivity and could access both management IPs for each 3PAR.

1. Log into the 3PAR SSMC.
2. Navigate to the Remote Copy Configurations screen.
3. Click Create configuration.
4. Select the primary 3PAR for the first system, and the secondary system for the second system.
5. Check the box for FC, and uncheck iP.
6. Select all four FC ports.
7. Click Create.
8. Once the configuration is complete, click Actions, and select Configure quorum witness.
9. Ensure that the right System pair and Target pair are selected.
10. Enter the IP of the virtual Quorum Witness appliance.
11. Click Configure.
12. Navigate to Remote Copy Groups.
13. Click Create group.
14. Select the primary array as the source.
15. Add a name for the group.
16. Select the RAID5 User and Copy CPG.
17. Select the secondary array as the target.
18. Select Synchronous for the mode.
19. Select the RAID5 User and Copy CPG.
20. Check the box for Path management and Auto failover.
21. Click Add source volumes.
22. Select all the desired source volumes (all volume for the first Oracle VM), and click Add.
23. Click Create.
24. Once all target volumes have synced, navigate to the secondary volumes.
25. Export these volumes to the same host as the primary volumes.
26. Your host should now show half the paths to the 3PAR storage devices as active, and the other half as standby.

Running the remote replication test

For our tests, we focused only on the behavior of each storage solution when there's a loss of primary storage connectivity to the host. We used a single host running a single Oracle VM running the same parameters as the first test. To simulate loss of storage connectivity, we created a separate zone configuration on each switch that removes the primary storage ports from the configuration. We used the historical volume performance data to show the I/O impact on each array. Additionally, we used the iostat info from SLOB to show impact on the VM.

1. Reboot the host and VM before each test.
2. Reboot everything, and power up the Oracle VM.
3. Wait 10 minutes.
4. Start esxtop on all hosts with the following script adjusting the number of iterations to match the test run time:

    ```
    esxtop -a -b -n 65 -d 60 > esxout.`date +"%Y-%m-%d_%H-%M-%S"`
    ```
5. Once all performance gathering tools have started, login to the Oracle VM as the `oracle` user, and navigate to /home/oracle/SLOB.
6. Run `./runit.sh 24` on each VM.
7. Allow SLOB to run for 20 minutes.
8. After 20 minutes, enable the zone configuration on the switch that disables the primary storage connection.
9. Once the storage connection goes down, observe the impact on the primary and secondary arrays.
10. Allow surviving SLOB runs and host performance gathering tools to complete.
11. Copy output data from the VMs and the hosts.
12. Use `Unishpere` to grab historical performance data from the VMAX array, and use `SSMC` to grab historical data from the 3PAR array.
13. As the `oracle` user on the Oracle VMs, use `rman` to delete the archive logs in between test runs.
14. We reported IOPS from the iostat output on the test VM where possible and the historical storage output when not available.

Appendix E: Oracle SPFILE

Database: ORCL

```
orcl.__data_transfer_cache_size=0
orcl.__db_cache_size=14696841216
orcl.__inmemory_ext_roarea=0
orcl.__inmemory_ext_rwarea=0
orcl.__java_pool_size=268435456
orcl.__large_pool_size=268435456
orcl.__oracle_base='/u01/app/oracle'#ORACLE_BASE set from environment
orcl.__pga_aggregate_target=5368709120
orcl.__sga_target=20266876928
orcl.__shared_io_pool_size=536870912
orcl.__shared_pool_size=4294967296
orcl.__streams_pool_size=0
*.audit_file_dest='/u01/app/oracle/admin/orcl/adump'
*.audit_trail='NONE'
*.compatible='12.2.0'
*.control_files='+DATA/ORCL/CONTROLFILE/current.260.947708117','+FRA/ORCL/CONTROLFILE/
current.256.947708117'
*.db_block_size=8192
*.db_cache_size=134217728
*.db_create_file_dest='+DATA'
*.db_file_multiblock_read_count=16
*.db_name='orcl'
*.db_recovery_file_dest='+FRA'
*.db_recovery_file_dest_size=590g
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=orclXDB)'
*.fast_start_mttr_target=180
*.filesystemio_options='setall'
*.local_listener='LISTENER_ORCL'
*.lock_sga=TRUE
*.log_archive_format='%t_%s_%r.dbf'
*.log_buffer=134217728#log buffer update
*.nls_language='AMERICAN'
*.nls_territory='AMERICA'
*.open_cursors=2000
*.parallel_max_servers=0
*.pga_aggregate_target=5368709120
*.processes=500
*.recyclebin='off'
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=19317m
*.shared_pool_size=4294967296
*.undo_retention=1
*.undo_tablespace='UNDOTBS1'
*.use_large_pages='only'
```

Appendix F: Benchmark parameters

We used the following slob.conf parameter settings. We changed the RUN_TIME parameter to the desired length based on which tests we conducted.

```
UPDATE_PCT=25
RUN_TIME=3600
WORK_LOOP=0
SCALE=50G
WORK_UNIT=64
REDO_STRESS=LITE
LOAD_PARALLEL_DEGREE=8

THREADS_PER_SCHEMA=4

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

#########################
#### Advanced settings:
#
# The following are Hot Spot related parameters.
# By default Hot Spot functionality is disabled (DO_HOTSPOT=FALSE).
#
DO_HOTSPOT=TRUE
HOTSPOT_MB=1200
HOTSPOT_OFFSET_MB=0
HOTSPOT_FREQUENCY=1

#
# The following controls operations on Hot Schema
# Default Value: 0. Default setting disables Hot Schema
#
HOT_SCHEMA_FREQUENCY=0

# The following parameters control think time between SLOB
# operations (SQL Executions).
# Setting the frequency to 0 disables think time.
#
THINK_TM_FREQUENCY=1
THINK_TM_MIN=.080
THINK_TM_MAX=.080
#########################

export UPDATE_PCT RUN_TIME WORK_LOOP SCALE WORK_UNIT LOAD_PARALLEL_DEGREE REDO_STRESS
export DO_HOTSPOT HOTSPOT_MB HOTSPOT_OFFSET_MB HOTSPOT_FREQUENCY HOT_SCHEMA_FREQUENCY THINK_TM_
FREQUENCY THINK_TM_MIN THINK_TM_MAX
```

This project was commissioned by Dell EMC.

PT Principled Technologies®

Facts matter.®