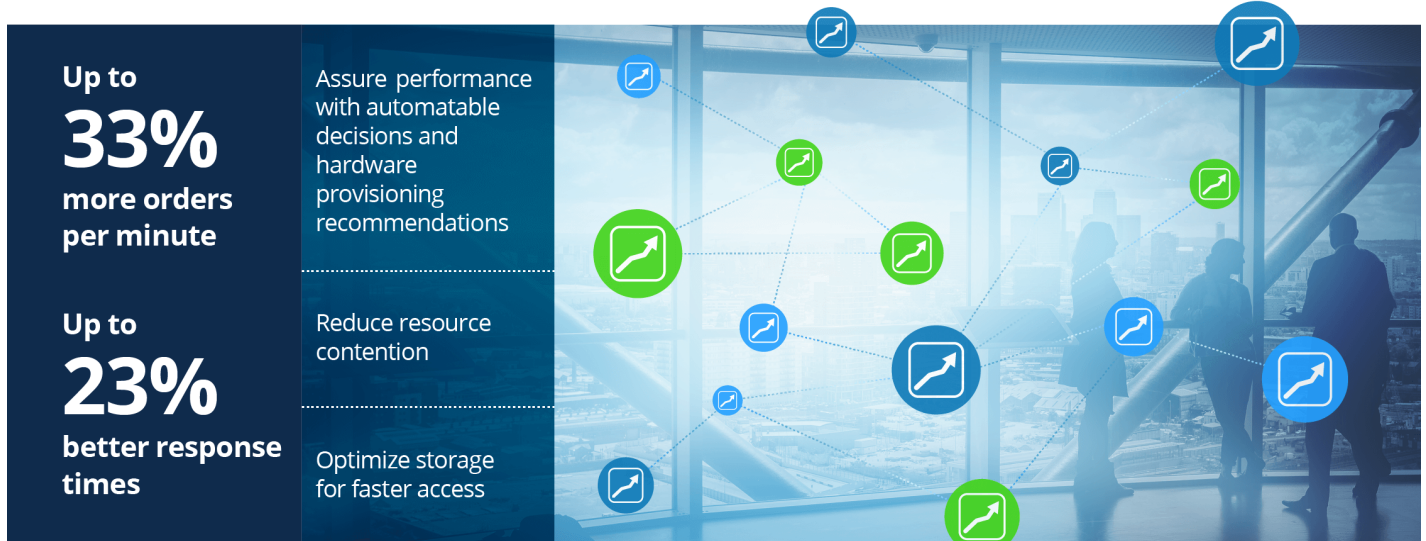


TURBONOMIC ENABLES GREATER PERFORMANCE FOR LARGE VIRTUAL ENVIRONMENTS

Boost VM production company-wide

Turbonomic Operations Manager decisions improved performance in a large-scale virtual environment*



*on VMware® vSphere® 6 compared to a solution without Turbonomic

As the number of virtual machines (VMs) in your infrastructure grows, so do the challenges. Delivering top performance for database VMs can be a tricky, time-consuming task for administrators. They have to sort through massive amounts of data and use trial and error to tweak resource allocation daily to get the best results.

Turbonomic Operations Manager is a platform that makes managing VMware vSphere VMs easier. Turbonomic understands the resource demands for all your applications, VMs, and containers, and makes decisions to assure they get the resources they need to perform. These decisions also ensure you use your resources wisely. We put the promise of Turbonomic to the test in the Principled Technologies labs, and we found that it did improve performance in our large-scale, 100 VM environment. When we used Turbonomic decisions, our cluster was able to handle up to 33 percent more database orders per minute with 23 percent better response times after one week of discovery.



THE IMPORTANCE OF OPTIMIZATION

If you buy a new dresser, do you fill only half of the drawers and then purchase additional furniture so that you can store all of your clothes? It may sound silly, but this is exactly what happens in datacenters across the globe with servers and storage. After making careful hardware selections based on ability and capacity, expensive resources can still be underutilized. An inability to assure application performance drives companies to purchase more hardware and expand the datacenter. This approach is costly and wasteful.

Even if your organization has done a good job of right-sizing your infrastructure for the VMs you need, it's still likely that processor, memory, and storage resources aren't being used efficiently and that performance is suffering from resource contention. Traditionally, administrators have had to look at logs of data to figure out where inefficiencies occur and then make those tweaks manually to try to optimize the environment for top performance. However, that process is time-consuming and isn't guaranteed to find every way to improve your infrastructure.

TURBONOMIC OPERATIONS MANAGER CAN HELP

Turbonomic Operations Manager assesses system-level resource demands in VMware vSphere environments automatically and makes automatable decisions that optimize performance. It discovers what application demands are and where storage, processor, and memory resources need to move to meet those demands with ease. In our testing, a pattern emerged. Utilizing the insight the Application Control Module and Storage Module granted, Turbonomic recognized the I/O demands of the SQL Servers in the test environment. Figure 1 shows an example of how Turbonomic intelligently placed virtual disks based on the performance of the underlying storage for some VMs. Throughout the week, Turbonomic would incrementally move more I/O heavy VMDKs to flash or high-speed disk-based storage, while shifting low-demand VMDKs to high capacity, low performance spindles. A key feature to the overall performance improvement over the solution without Turbonomic was that the Turbonomic decision engine continuously managed the tradeoff between performance and capacity improvements against the current workload. This means that storage and VM migrations happened one at a time, minimizing additional load on the storage and network subsystems.

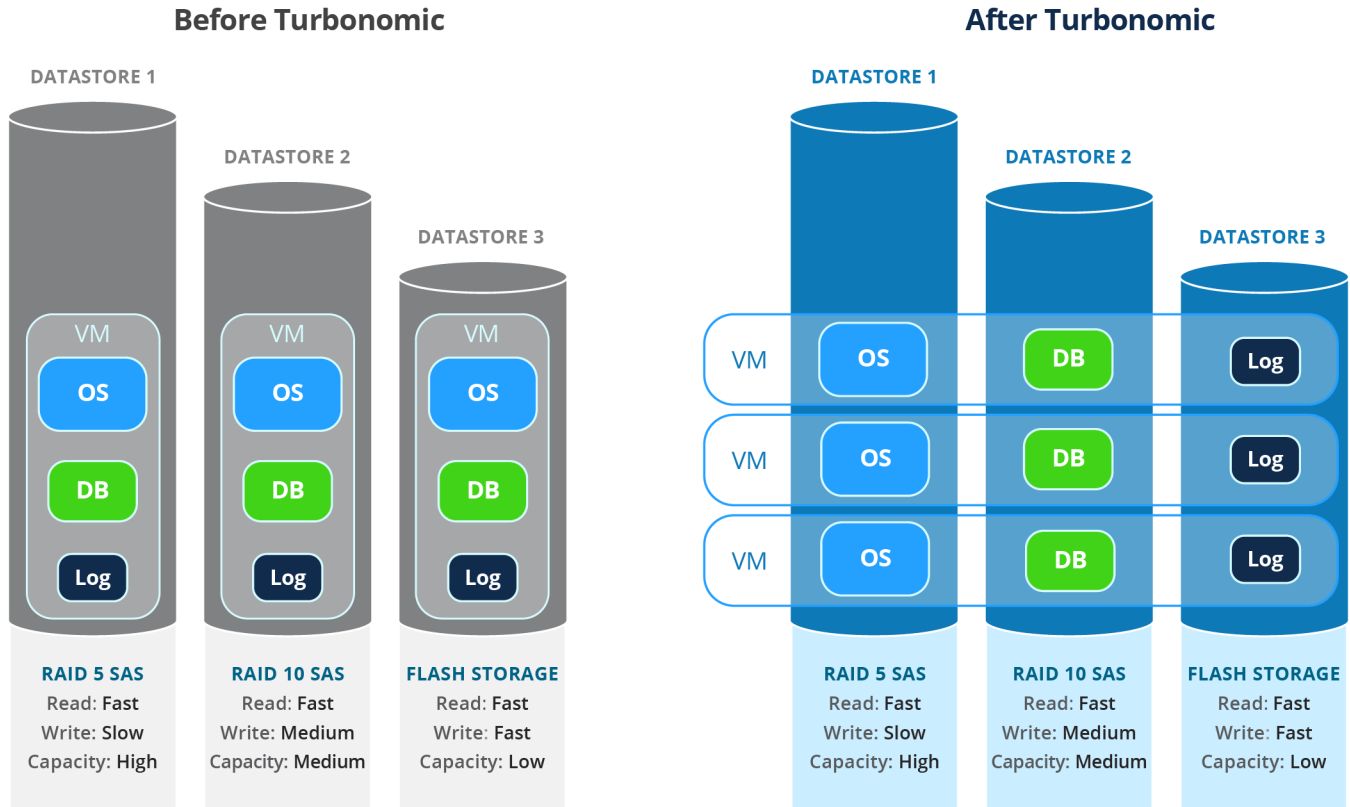
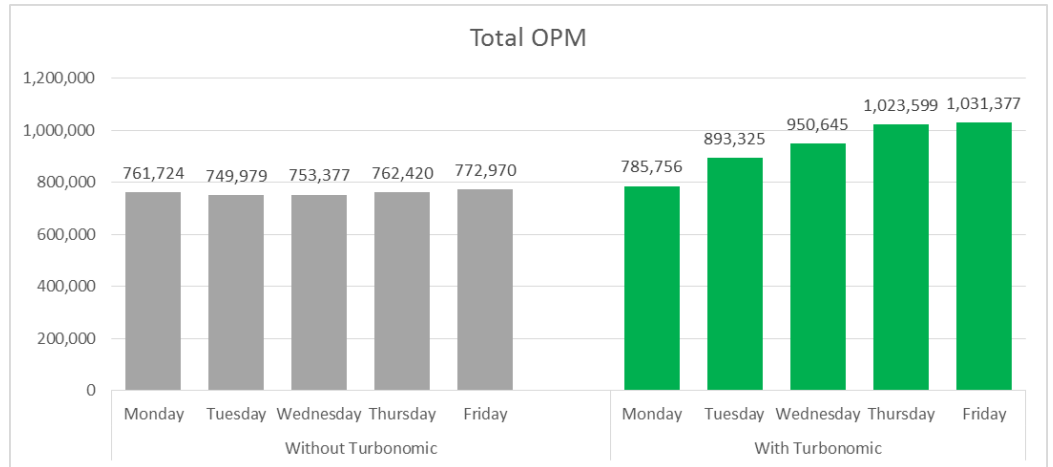


Figure 1: How Turbonomic helps optimize performance.

As Figure 2 shows, using the Turbonomic decision algorithms to place the SQL workload across existing storage, hot-add vCPUs, and determine and provision additional hardware increased the total OPM output of our 100-VM infrastructure by 33 percent. We performed Week 1 testing without Turbonomic. During this time, Storage DRS successfully rebalanced storage allocation for more efficient utilization of datastore resources, while incrementally improving performance through I/O monitoring and analysis.

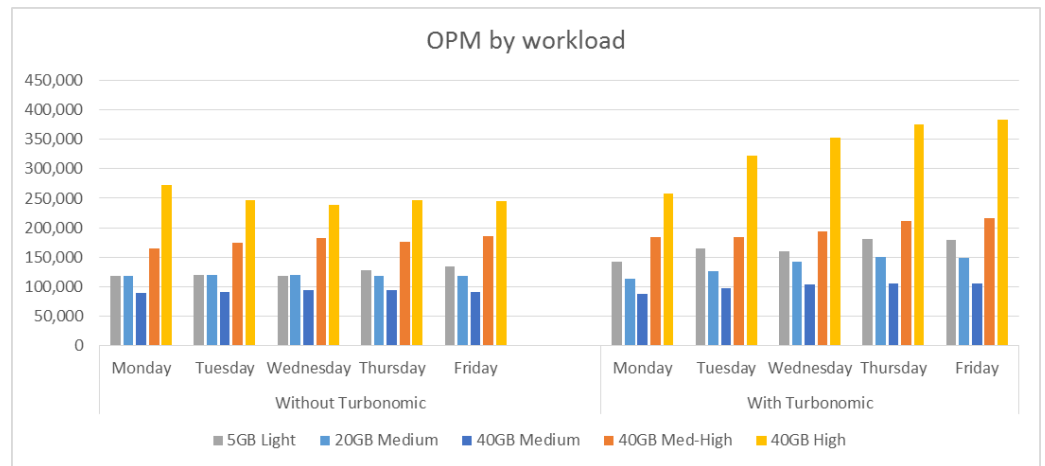
During Week 2 (in green), we enabled Turbonomic automation and manually enacted Turbonomic-prescribed measures which included adding lower-performance RAID 10 SAS storage in the cluster and two additional hosts. Turbonomic optimized the workload across these new resources automatically, placing the VMs equally on the identically configured blades. Turbonomic also recognized the storage as lower performing, and selectively migrated only enough VMDKs to lighten the performance burden on the original storage pools without degrading the performance of the relocated VMDKs.

Figure 2: Using Turbonomic to control and implement virtual and physical resource increases to our large-scale environment resulted in OPM increases of up to 33 percent over the week.



In our tests, we ran six different workloads on the VMs: 20GB light, 20GB medium, 40GB medium, 40GB med-high, and 40GB high. Figure 3 shows the performance by workload over the course of the week.

Figure 3: Using Turbonomic and implementing its suggested configuration changes and provisioning instructions increased performance across all workloads over the week.



A faster system with less latency, or wait time, goes a long way to improving user satisfaction with applications that run on that server. As Figures 4 and 5 show, application latency decreased by up to 23 percent on average after implementing a week’s worth of decisions from Turbonomic, which included automated actions such as hot-adding vCPUs and memory to resource-constrained VMs, increasing SQL transaction log sizes, granular VMDK migrations based on specific workload demands, and provisioning additional disks and hosts when requested by Turbonomic.

Figure 4: Turbonomic decreased application latency, which means that users get a faster response when they complete their tasks.

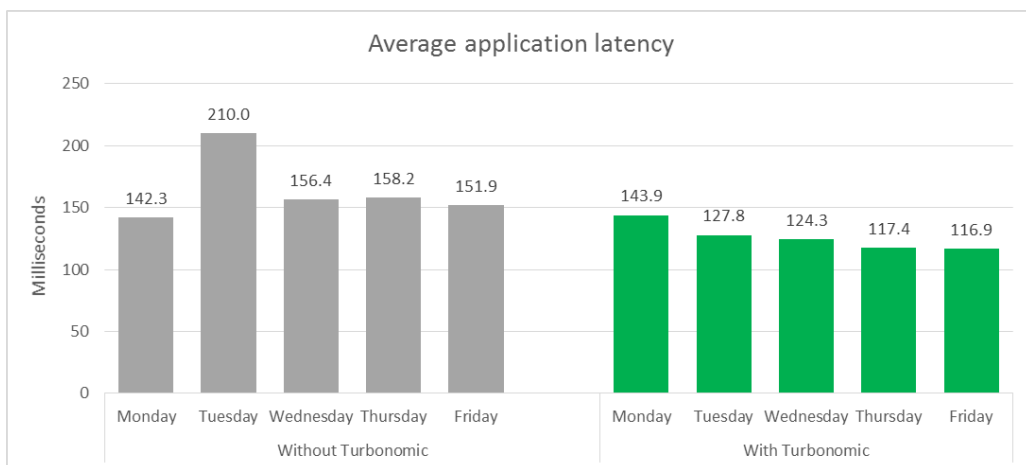
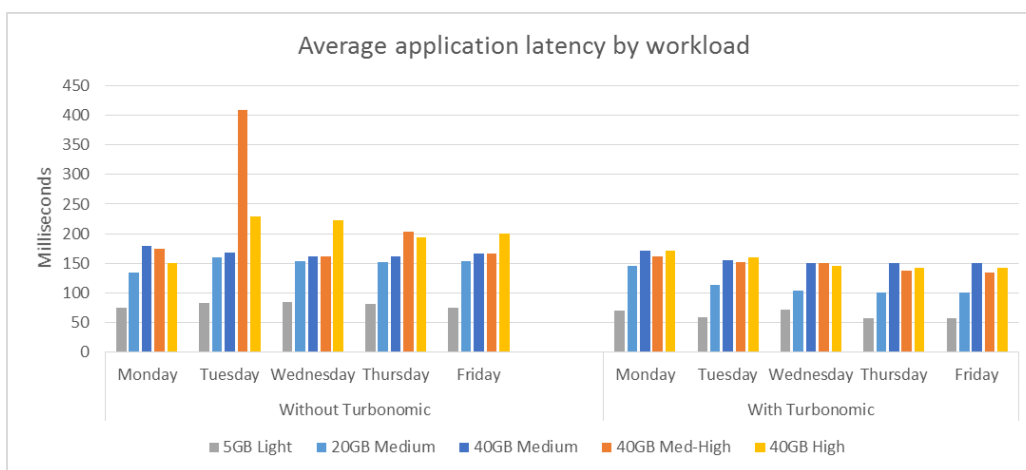


Figure 5: Turbonomic decreased application latency, which means that users get a faster response when they complete their tasks.



WHAT KINDS OF CONTROL DID TURBONOMIC ENABLE?

With the integration of control modules into Turbonomic 5, administrators now have a variety of options to extend control across their environment. Each module provides an expanded understanding of the supply chain of resources that applications need to perform, enabling decisions that are more comprehensive and the ability to prevent future problems. In our testing of this environment, we found that storage was the primary bottleneck on the performance, so most decisions targeted storage issues.

At the most basic level, Turbonomic decides on moving VMs to less-congested storage. Beyond that, it provides alternatives, such as moving a datastore to a different pool of disks and provisioning more disks to a pool to alleviate capacity and performance choke points. A key feature of the Turbonomic automated host and storage migration is the decision of which VMs to migrate. Rather than migrate a busy VM from a congested host or storage volume, which will increase latency for the active

VM, Turbonomic chooses less-utilized VMs, which are less likely to suffer from the migration process.

Figures 6 and 7 show a timeline of Turbonomic automated actions including additional storage and host provisioning decisions. The first day, we allowed Turbonomic to provide decisions based on the previous week's results, without providing any additional hardware. Turbonomic spent the day migrating VMs to more appropriate storage (see Figure 1), and provided additional vCPU resources to VMs using VMware hot-add functionality. The SQL application control module also provided decisions to increase the vMemory allocation to several VMs, which we performed manually via hot-add.

On the second day of testing, the Turbonomic decision engine requested additional storage resources due to disk latency and capacity. In response to the request, we provisioned a new RAID 10 pool of eight 15K SAS disks, which increased the number of disks of this performance tier by about 10 percent. Turbonomic correctly analyzed the performance of the storage and migrated disks from several VMs to help alleviate I/O issues on the most heavily used VM volumes to help reduce overall congestion on the storage, but did not perform a larger migration to prevent overloading the performance capabilities of the new disks. As the workload progressed, the Turbonomic optimizations of the previous day meant that the workload was further processor-constrained, and began hot-adding vCPUs to compensate.

Later that day, Turbonomic requested more storage resources because of disk latency, and we provisioned a new RAID 10 pool of ten 10K SAS disks, which doubled the number of disks of the lowest performance tier. Again, Turbonomic decided to migrate a small number of VMs when necessary, preventing moving a high-performance workload to slower storage, while avoiding overloading the new disks (Pools 7 and 8; see Figure 13).

Day 3 brought the first host provisioning decision. We added an identical host to the cluster to match the first two. Turbonomic decided on inactive VMs to migrate to the new host one at a time, minimizing the effect of the migrations on the network. This is in contrast to other solutions, which have a tendency to mass-migrate VMs when a new resource is presented. While this form of resource distribution can be effective, it may choke active resources in the environment, causing temporary performance issues.

Day 4 brought another host provisioning decision, and a series of migrations to balance the workload across hosts. Once the initial migrations were complete, Turbonomic reassessed the relative load of the four hosts of the cluster, and migrated VMs to underutilized hosts as the workload progressed through the remainder of the day.

On Day 5, the environment approached equilibrium with the workload. As Figures 2 through 5 show, the workload optimizations reached a plateau, and Turbonomic no longer requested further disk pools be allocated to the environment.

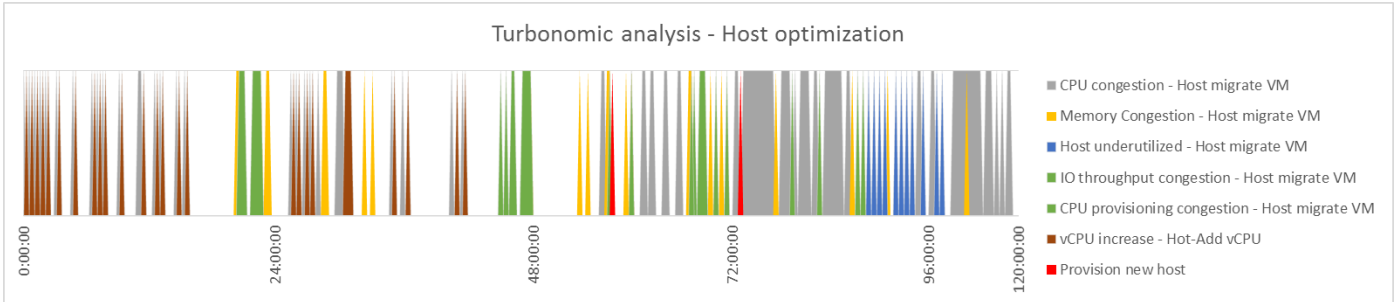


Figure 6: Turbonomic decision timeline.

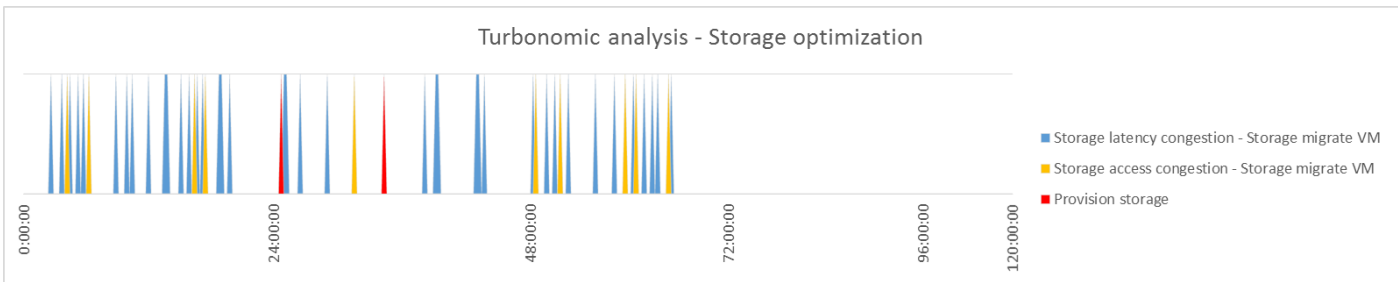


Figure 7: Turbonomic decision timeline.

Throughout the test, the application and fabric control modules also had decisions for how to improve the efficiency of our environment. Delving into the SQL performance data, Turbonomic decided that increasing the transaction log size on high-usage databases could improve performance, instead of a more costly, brute-force method of only increasing vCPU or vMemory allocation. The fabric module recognized that one of the server links in a port channel was underutilized, and suggested removing it from the group—a move which would not affect performance, but instead save on expensive per-port fabric licenses.

CONCLUSION

There's no reason to leave anything on the table when it comes to assuring application performance at scale. Turbonomic Operations Manager can help you control resource and storage allocation in a simple, straightforward way and ensure that you're getting even more out of your databases. We found that adding Turbonomic to a virtual environment enabled us to better use existing resources, and showed us when additional hardware was needed to reach peak performance on our workload. Implementing these decisions increased orders per minute by up to 33 percent and delivered 23 percent better response times over our week of testing.

For users, Turbonomic can enable better response times, which means less waiting and frustration. This might mean that a customer can check out faster in an online store or that an employee can pull up a file instantaneously, both of which mean happier people.

For administrators, Turbonomic can make day-to-day life easier because it removes a time-consuming repetitive task from their workday. Turbonomic can find ways to optimize resources more efficiently, so it takes less time to get better results and frees administrators up to focus on more important projects.

For organizations, Turbonomic can help cut costs because it enables better performance from hardware you already have. That way, you don't have to invest in hardware and expand before you really need to, because you're maximizing what the resources you have can do.

If you're looking to assure application performance and get even more out of your VMware vSphere virtual environment, Turbonomic Operations Manager can help optimize your environment and make CEOs, administrators, and users happy.

APPENDIX A – ABOUT THE COMPONENTS

About Turbonomic Operations Manager

According to Turbonomic, “Turbonomic’s Application Performance Control system enables customers to guarantee Quality of Service for any application while maximizing resource utilization of any infrastructure.

“Turbonomic’s patented technology continuously matches any application workload demand to any infrastructure supply. With this unique, real-time capability, Turbonomic is the only technology capable of controlling and maintaining an environment in a healthy state.”

Visit the Turbonomic website at www.turbonomic.com to learn more.

About DVD Store Version 2.1

To create our real-world ecommerce workload, we used the DVD Store Version 2.1 benchmarking tool. DS2 models an online DVD store, where customers log in, search for movies, and make purchases. DS2 reports these actions in orders per minute that the system could handle, to show what kind of performance you could expect for your customers. The DS2 workload also performs other actions, such as adding new customers, to exercise the wide range of database functions you would need to run your ecommerce environment.

For more details about the DS2 tool, see www.delltechcenter.com/page/DVD+Store.

APPENDIX B – SYSTEM CONFIGURATION INFORMATION

Figure 8 provides detailed configuration information for the test systems. All four Cisco UCS® B200 M3 blades under test were configured identically.

System	Cisco UCS B200 M3 Blade Server
Enclosure	
Blade enclosure	Cisco UCS 5108
Power Supplies	
Total number	4
Wattage of each (W)	2,500
Cooling fans	
Total number	8
General	
Number of processor packages	2
Number of cores per processor	8
Number of hardware threads per core	2
System power management policy	High Performance
CPU	
Vendor	Intel®
Name	Xeon®
Model number	E5-2650
Stepping	6
Socket type	LGA2011
Core frequency (GHz)	2.0
Bus frequency	8.0 GT/s
L1 cache	512 KB
L2 cache	2 MB
L3 cache	20 MB
Platform	
Vendor and model number	Cisco UCS B200 M3
Motherboard model number	73-14689-04
BIOS name and version	Cisco® B200M3.2.2.4a.0.041720151158
BIOS settings	Default
Memory module(s)	
Total RAM in system (GB)	384
Vendor and model number	Hynix® UCS-MR-1X162RZ-A
Type	DDR3-1866 PC3-14900
Speed (MHz)	1,866
Speed running in the system (MHz)	1,866
Size (GB)	16
Number of RAM module(s)	24
Chip organization	Double-sided

System	Cisco UCS B200 M3 Blade Server
Operating system	
Name	VMware vSphere 6.0
Build number	2143827
Language	English
RAID controller	
Vendor and model number	LSI Logic® MegaRAID SAS 2004 ROMB
Firmware version	20.12.1-0206 4.37.00 NA
Network adapter	
Vendor and model number	Cisco UCS VIC 1240

Figure 8: System configuration information for the test systems.

APPENDIX C – HOW WE TESTED

To test Turbonomic’s ability to improve performance, we created a two-node cluster in VMware vCenter™ 6.0 and evenly distributed an OLTP-based workload across both hosts, and used two nodes for an infrastructure cluster for the test environment. Additionally, we created two reserve datastores and configured two reserve ESXi hosts to add to the environment if Turbonomic indicated more raw capacity was required.

We configured two datastores per disk pool on the Fibre Channel storage to simulate a typical environment involving contention due to multiple datastores per pool.

To simulate a realistic environment of a large database farm, we deployed test clients which requested data from the Microsoft® SQL servers running several different workloads on three different database sizes. Each Microsoft SQL Server® was configured with separate OS, transaction log, and database volumes. We used DVD Store as the benchmark tool to generate the workload and evaluate the performance of the environment. The combined workload ran for a total of eight hours, with five VMs from each workload segment running for two hours.

During DRS testing, we configured DRS to auto-accept level 1, 2, and 3 suggestions, while Storage DRS was configured to automatically run, including I/O optimizations at the default levels.

We configured Turbonomic to automatically relocate VMs to hosts or storage, and automatically hot-add vCPU and memory. All other Turbonomic recommendations were manually implemented during testing.

Test bed configuration

Figure 9 shows a diagram of our test bed. The blade chassis was connected to an 8Gb Fibre Channel storage array.

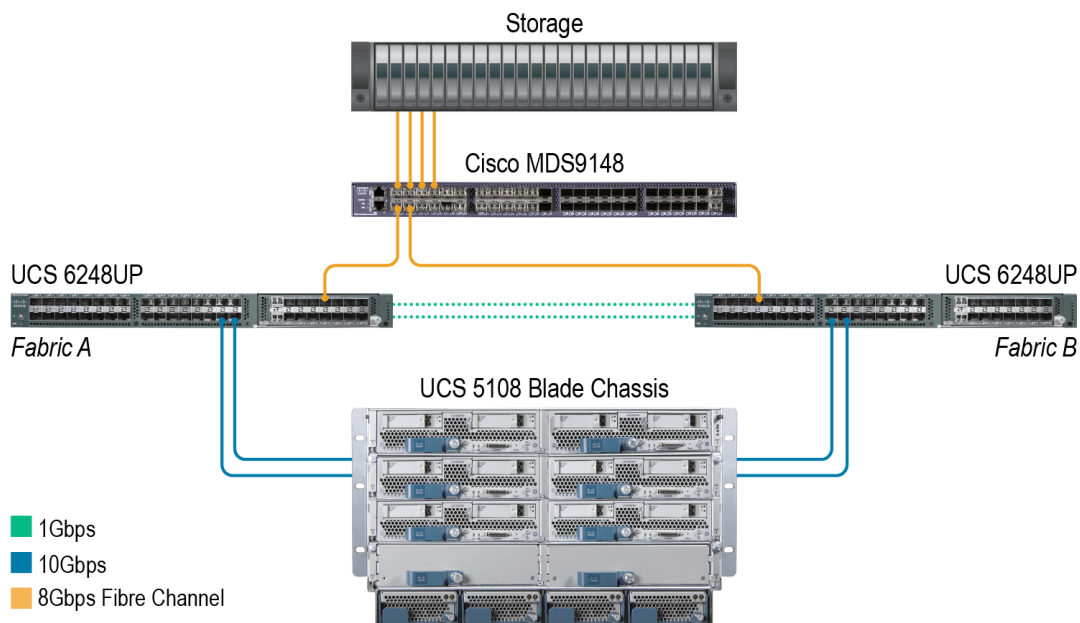


Figure 9: Test bed diagram.

Configuring the test bed

Installing VMware ESXi™ 6.0 on the infrastructure servers and servers under test

1. Connect to the UCS server via KVM, and attach the installation media.
2. Boot the server.
3. At the VMware Installer screen, press Enter.
4. At the EULA screen, press F11 to Accept and Continue.
5. Under Storage Devices, select the appropriate virtual disk, and press Enter.
6. For keyboard layout, select US, and press Enter.
7. Enter the root password twice, and press Enter.
8. To start installation, press F11.
9. After the server reboots, press F2, and enter root credentials.
10. Select Configure Management Network, and press Enter.
11. Select the appropriate network adapter, and click OK.
12. Log into the server using the VMware vSphere client.
13. Select the Configuration tab, and click Networking.
14. Click Add Networking...
15. Create the management network on VLAN 102.
16. Click the Configuration tab, and select Time configuration.
17. Click DNS and Routing, and confirm the settings.

Setting up Windows Server® VMs

Create VMs for DC1, Infra-SQL, and vCenter for the infrastructure. The same procedure is used to create the test VMs. See Figure 11 for configuration values.

1. Log into the ESX server with the VMware vSphere client.
2. In the VMware vSphere client, under Basic Tasks, select Create a new virtual machine.
3. Choose Custom, and click Next.
4. Assign a name to the virtual machine, and click Next.
5. Select the host, and click Next.
6. Select the appropriate storage, and click Next.
7. Choose Virtual Machine Version 11, and click Next.
8. Choose Windows®, select Microsoft Windows Server 2012 R2 (64-bit), and click Next.
9. For CPUs, select the correct number of vCPUs based on the VM role, and click Next.
10. Choose the amount of memory based on the VM role, and click Next.
11. Click 1 for the number of NICs, select VMXNET3, connect to the PRIV-NET port group, and click Next.
12. Leave the default virtual storage controller, and click Next.
13. Keep the default virtual device node (0:0), and click Next.
14. Connect the VM virtual CD-ROM to the Microsoft Windows 2012 R2 installation disk.
15. Click Finish.
16. Right-click the vCenter VM, and click Edit settings.
17. Start the VM.

Installing Microsoft Windows Server 2012 R2

1. Choose the language, time and currency, and keyboard input. Click Next.
2. Click Install Now.
3. Select Windows Server 2012 R2 Datacenter, and click Next.
4. Accept the license terms, and click Next.
5. Click Custom.
6. Click the Disk, and click Next.

7. After the installation completes, enter the administrator password twice, and click Finish.

Configuring the operating system

1. Enable remote desktop access.
2. Change the hostname, and reboot when the installer prompts you.
3. Set up networking:
 - a. Click Start→Control Panel, right-click Network Connections, and select Open.
 - b. Right-click the management traffic NIC, and select Properties.
 - c. Select TCP/IP (v4), and select Properties.
 - d. Set the IP address and subnet for the NIC, which will handle management traffic, and click OK.
 - e. Click OK, and click Close.
4. Install all available Windows updates. Restart as necessary.
5. Once DC1 is configured as an Active Directory® server, join all other VMs to the AD before continuing.

Installing Active Directory and DNS services

1. Launch Server Manager.
2. Click Manage→Add Roles and Features.
3. At the Before you begin screen, click Next.
4. At the Select installation type screen, leave Role-based or feature-based installation selected, and click Next.
5. At the Server Selection Screen, select the server from the pool, and click Next.
6. At the Select Server Roles screen, select Active Directory Domain Services. Click Add Features when prompted, and click Next.
7. At the Select Features screen, click Next.
8. At the Active Directory Domain Services screen, click Next.
9. At the Confirm installation selections screen, check Restart the destination server automatically if required, and click Install.

Configuring Active Directory and DNS services

1. After the installation completes, a screen should pop up with configuration options. If not, click the Tasks flag in the upper-right section of Server Manager.
2. Click Promote this server to a Domain Controller.
3. At the Deployment Configuration screen, select Add a new forest. In the Root domain name field, type `test.local` and click Next.
4. At the Domain Controller Options screen, leave the default values, and enter a password twice.
5. Click Next four times to accept default settings for DNS, NetBIOS, and directory paths.
6. At the Review Options screen, click Next.
7. At the Prerequisites Check dialog, allow the check to complete. If there are no relevant errors, check Restart the destination server automatically if required, and click Install.
8. When the server restarts, log on using `test\Administrator` and the specified password.

Configuring the Windows Time Service on DC1

To ensure reliable time, we pointed our Active Directory server to a local NTP server.

1. Open a command prompt.
2. Type the following:

```
W32tm /config /syncfromflags:manual /manualpeerlist:"<ip address of a NTP server>"
W32tm /config /reliable:yes
W32tm /config /update
W32tm /resync
Net stop w32time
Net start w32time
```

Installing and configuring Microsoft SQL Server 2014

1. Open the console for the VM, or connect to the VM with RDP.
2. Mount the installation ISO to the VM.
3. Click to run SETUP.EXE. If Autoplay does not begin the installation, navigate to the SQL Server 2014 DVD, and double-click.
4. If the installer prompts you with a .NET installation prompt, click Yes to enable the .NET Framework Core role.
5. In the left pane, click Installation.
6. Click New SQL Server stand-alone installation or add features to an existing installation.
7. At the Setup Support Rules screen, wait for the rule check to complete. If there are no failures or relevant warnings, click OK.
8. Select Specify a free edition, and from the drop-down menu, select Evaluation. Click Next.
9. Click the checkbox to accept the license terms, and click Next.
10. If no failures are displayed after the setup support files are installed, click Next.
11. At the Setup Role screen, choose SQL Server Feature Installation.
12. At the Feature Selection screen, select Database Engine Services, Full-Text and Semantic Extractions for Search, Client Tools Connectivity, Client Tools Backwards Compatibility, Management Tools – Basic, and Management Tools – Complete. 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 Disk Space Requirements screen, click Next.
16. At the Server Configuration screen, check that NT AUTHORITY\SYSTEM is selected for SQL Server Agent and SQL Server Database Engine. Click Next.
17. Assuming there are no failures, click Next.
18. At the Database Engine Configuration screen, select Mixed Mode.
19. Enter and confirm a password for the system administrator account.
20. Click Add Current user. This may take several seconds.
21. Click the Data Directories tab, and select the default data locations and log locations.
22. Click Next.
23. At the Error and usage reporting screen, click Next.
24. At the Installation Configuration Rules screen, check that there are no failures or relevant warnings, and click Next.
25. At the Ready to Install screen, click Install.
26. After installation completes, click Close.
27. Download and install any available updates for Microsoft SQL Server 2014.

Setting up an ODBC DSN for vCenter

1. Create new databases on Infra-SQL in Microsoft SQL Management Studio named `vcenter` and `vupdate`
2. Click Options, change the recovery model from full to simple on both databases, and click OK.
3. Log into vCenter as `TEST\administrator`
4. From the desktop of the vCenter server, select Start→Run, and type `odbcad32.exe`. Press Enter.
5. Click the system DSN tab.
6. Click Add.
7. Click SQL Server Native Client 11.0, and click Finish.
8. In the Create a New Data Source to SQL Server text box, type the connection name `vcenter`
9. For Server, select SQL, and click Next.
10. Change authentication to With SQL Server authentication using a login ID and password entered by the user, type `sa` as the Login ID, use the password you defined in SQL server setup for the SA account, and click Next.
11. Select Change the default database to choose vCenter from the pull-down menu, and click Next.

12. Click Finish.
13. To confirm correct configuration, click Test Data Source...
14. To create the vCenter ODBC connection, click OK.
15. Repeat the procedure to create a connection for the vupdate database.

Installing VMware vCenter 6.0 on Windows Server 2012 R2

1. From the VMware vCenter 6.0 install media, double-click Autorun.exe.
2. To start the install wizard, click Install.
3. Click Next.
4. Check the box to accept the license agreement, and click Next.
5. Leave the default radio button selection (vCenter Server and Embedded Platform Services Controller), and click Next.
6. Provide the correct system FQDN, and click Next.
7. Select the radio button to create a new Single Sign-on domain, and fill in all necessary information. Click Next.
8. Choose to run the vCenter service under a domain account or the Windows Local System Account. If a domain account, provide the username and password. Click Next.
9. Provide database credentials to use the database created previously. Click Next.
10. Leave all ports configured to their default, and click Next.
11. Choose a location for vCenter to install files, and choose a directory for vCenter data. Click Next.
12. Verify all settings on the Ready to install screen, and click Install.
13. After install has completed, select Finish.
14. On the installer, click Exit.
15. Restart the server.
16. Navigate to the IP address of the vCenter installation in a web browser, and log into the vCenter server using the domain account `TEST\administrator`
17. Right-click the root of vCenter, and click New Data center.
18. Name the new datacenter `datacenter`
19. Add all ESX servers under test to the datacenter.

Installing Internet Information Services

1. Click Start→Administrative Tools→Server Manager.
2. On the left pane, click Roles.
3. Click Add Roles.
4. Click the Application Server checkbox.
5. When the Add features required for Web Server (IIS)? screen appears, click Add Features.
6. Select .NET Framework 3.5 and 4.5, and add ASP.NET 4.5.
7. Click Next.
8. Click Next.
9. At the Select Role Services page for Web Server (IIS), click Next.
10. Click Install.
11. Click Close.

Configuring Internet Information Services

1. Copy the pre-configured ds2web folder containing the HTML and .cs files for the DS2 Web application to C:\
2. Open Internet Information Services (IIS) Manager.
3. Expand the server tree, and disable the Default Web Site.
4. Add a website named `ds2`
5. Set the Physical path to `C:\ds2web`
6. Click OK.
7. Add a Virtual Directory, and point the virtual path `/ds2` to the root application.

Configuring the database workload client

For our testing, we used a virtual client for the IIS/Microsoft SQL Server client. To create this client, we installed Windows Server 2012 R2, assigned a static IP address, and installed .NET 4.5.

Configuring the database

Data generation overview

We generated the data using the Install.pl script included with DVD Store version 2.1 (DS2), providing the parameters for our 5GB, 20GB, and 40GB database sizes and the Microsoft SQL Server 2014 platform. We ran the Install.pl script on a utility system running Linux®. The Install.pl script also generated the database schema.

After processing the data generation, we transferred the data files and schema creation files to a Windows-based system running SQL Server 2012. We built the databases in SQL Server 2012, and then performed a full backup, storing the backup file on a network share for quick access. We used that backup file to restore the servers between test runs.

The only modification we made to the schema creation scripts were the specified file sizes for our databases. We explicitly set the file sizes higher than necessary to ensure that no file-growth activity would affect the outputs of the test. Besides this file size modification, the database schema was created and loaded according to the DVD Store documentation. Specifically, we followed the steps below:

1. We generated the data and created the database and file structure using database creation scripts in the DS2 download. We made size modifications specific to our databases and the appropriate changes to drive letters.
2. We transferred the files from our Linux data generation system to a Windows system running SQL Server.
3. We created database tables, stored procedures, and objects using the provided DVD Store scripts.
4. We set the database recovery model to bulk-logged to prevent excess logging.
5. We loaded the data we generated into the database. For data loading, we used the import wizard in SQL Server Management Studio. Where necessary, we retained options from the original scripts, such as Enable Identity Insert.
6. We created indices, full-text catalogs, primary keys, and foreign keys using the database-creation scripts.
7. We updated statistics on each table according to database-creation scripts, which sample 18 percent of the table data.
8. On the SQL Server instance, we created a ds2user SQL Server login using the following Transact-SQL (T-SQL) script:

```
USE [master]
GO
CREATE LOGIN [ds2user] WITH PASSWORD=N'',
    DEFAULT_DATABASE=[master],
    DEFAULT_LANGUAGE=[us_english],
    CHECK_EXPIRATION=OFF,
    CHECK_POLICY=OFF
GO
```

9. We set the database recovery model back to full.
10. We created the necessary full text index using SQL Server Management Studio.
11. We created a database user and mapped this user to the SQL Server login.

12. We then performed a full backup of the database. This backup allowed us to restore the databases to a pristine state relatively quickly between tests.

Figures 10 through 12 show our initial file size modifications.

Logical name	Filegroup	Initial size (MB)
Database files (5GB database)		
Primary	PRIMARY	4
cust1	DS_CUST_FG	2,560
cust2	DS_CUST_FG	2,560
ind1	DS_IND_FG	1,536
ind2	DS_IND_FG	1,536
ds_misc	DS_MISC_FG	1,536
orders1	DS_ORDERS	1,536
orders2	DS_ORDERS	1,536
Log files		
ds_log	Not applicable	10,240

Figure 10: Our initial file size modifications.

Logical name	Filegroup	Initial size (MB)
Database files (20GB database)		
Primary	PRIMARY	5
cust1	DS_CUST_FG	6,144
cust2	DS_CUST_FG	6,144
cust3	DS_CUST_FG	6,144
cust4	DS_CUST_FG	6,144
ind1	DS_IND_FG	2,891
ind2	DS_IND_FG	2,891
ind3	DS_IND_FG	2,891
ind4	DS_IND_FG	2,891
ds_misc	DS_MISC_FG	200
orders1	DS_ORDERS	3,072
orders2	DS_ORDERS	3,072
orders3	DS_ORDERS	3,072
orders4	DS_ORDERS	3,072
Log files		
ds_log	Not applicable	18,141

Figure 11: Our initial file size modifications.

Logical name	Filegroup	Initial size (MB)
Database files (40GB database)		
Primary	PRIMARY	5
cust1	DS_CUST_FG	12,288
cust2	DS_CUST_FG	12,288
cust3	DS_CUST_FG	12,288
cust4	DS_CUST_FG	12,288

Logical name	Filegroup	Initial size (MB)
ind1	DS_IND_FG	5,826
ind2	DS_IND_FG	5,826
ind3	DS_IND_FG	5,826
ind4	DS_IND_FG	5,826
ds_misc	DS_MISC_FG	269
orders1	DS_ORDERS	6,144
orders2	DS_ORDERS	6,144
orders3	DS_ORDERS	6,144
orders4	DS_ORDERS	6,144
Log files		
ds_log	Not applicable	35,437

Figure 12: Our initial file size modifications.

Configuring storage

Figure 13 provides information on the configuration of the storage for the vSphere clusters and the mount points.

Storage, volumes, and storage pools	
Enclosure	
Storage pool 1	
RAID level	RAID 5 (8+1)
Drive size (GB)	200
Drive RPM or SSD	SSD
Number of drives	9
Drive type	6Gb SAS 3.5"
Storage pool 2	
RAID level	RAID 10
Drive size (GB)	300
Drive RPM or SSD	15K
Number of drives	16
Drive type	6Gb SAS 3.5"
Storage pool 3	
RAID level	RAID 5 (4+1)
Drive size (GB)	600
Drive RPM or SSD	10K
Number of drives	10
Drive type	6Gb SAS 3.5"
Storage pool 4	
RAID level	RAID 10
Drive size (GB)	300
Drive RPM or SSD	15K
Number of drives	16
Drive type	6Gb SAS 3.5"

Storage, volumes, and storage pools	
Storage pool 5	
RAID level	RAID 10
Drive size (GB)	300
Drive RPM or SSD	15K
Number of drives	24
Drive type	6Gb SAS 3.5"
Storage pool 6	
RAID level	RAID 10
Drive size (GB)	300
Drive RPM or SSD	15K
Number of drives	32
Drive type	6Gb SAS 3.5"
Storage pool 7	
RAID level	RAID 10
Drive size (GB)	300
Drive RPM or SSD	15K
Number of drives	8
Drive type	6Gb SAS 3.5"
Storage pool 8	
RAID level	RAID 10
Drive size (GB)	600
Drive RPM or SSD	10K
Number of drives	10
Drive type	6Gb SAS 3.5"

Figure 13: Configuration information for the Generic NAS solution.

Configuring Turbonomic

Installing Turbonomic

1. Install the appliance OVF using vCenter.
 - a. Select File→Deploy OVF Template.
 - b. Browse to the location of the OVA file.
 - c. Continue through the wizard, and accept the license agreement.
 - d. Name the new appliance VM.
 - e. Select the destination datastore, and choose the provision method.
 - f. Map the VM network to the test network.
 - g. Select Power on after deployment, and click Finish.
2. Open a console to the Turbonomic appliance.
3. Log into the appliance with the default `ipsetup/ipsetup` credentials.
4. Set the static IP address for the appliance.
5. Confirm the change.
6. Log in as `root/Turbonomic`

7. Execute `yast timezone` and set the correct time zone and system time.
8. Connect to the IP address via web browser.
9. Log in as `administrator/administrator`
10. Install the Turbonomic license.
11. Navigate to Admin→User Configuration.
12. Enter the Active Directory Domain and Server information, and click Apply.
13. Navigate to Admin→Maintenance.
14. Select Check for software updates, and apply any available updates.

Configuring Admin

1. Navigate to Admin→Target Configuration.
2. Select Add, and select Hypervisor.
3. Enter the host name/IP address and authentication credentials for the vCenter, and click Apply.
4. Select Add, and select Storage.
5. Enter the IP address of the SMI-S provider and authentication credentials for the block level storage, and click Apply.
6. Select Add, and select Fabric.
7. Enter the IP address of the Cisco UCS chassis and authentication credentials, and click Apply.
8. Select Add, and select Database.
9. Select the scope of SQL VMs under test and enter authentication credentials, and click Apply.
10. Allow at least 15 minutes for the Turbonomic appliance to discover the infrastructure.

Configuring Policy

1. Under Action→VM→Virtual Machines by PM cluster, enable the overrides for Move, Storage Move, and Resize up, and set the value to Automated.
2. Under Discovery→Application Discovery→Application Credentials→Virtual Machines, override the Username and Password attributes, and enter the administrator authentication credentials.
3. Under Analysis, check the box Ignore Hyperthreading.
4. Under Analysis→Utilization Constraints, change the Storage Amount and IOPS thresholds to 80 percent, Host CPU Utilization to 60 percent, and Host Memory to 80 percent.

Test start and run times

We ran the workloads in staggered groups in order to start and record all performance counters for this report. The specific testing parameters we used for each benchmark are included in the setup sections, and we describe specifics for launching these tests below.

About running the DVD Store tests

We created a series of batch files, SQL scripts, and shell scripts to automate the complete test cycle. DVD Store outputs an orders-per-minute metric, which is a running average calculated through the test. In this report, we report the last OPM reported by each client/target pair.

Each complete test cycle consisted of the general steps listed below.

1. Reboot the ESX hosts.
2. Clean up prior outputs from the target system and the client driver system.
3. Drop the database from the target.
4. Restore the database on the target.
5. Wait for a ping response from the server under test (the hypervisor system), the client system, and the target.

6. Let the test server idle for 5 minutes.
7. Start the DVD Store driver on the client.
8. We used the following DVD Store parameters for testing (5GB database example):

```
ds2sqlserverdriver.exe --target=<target_IP> --ramp_rate=100 --run_time=120 -
-n_threads=8 --db_size=5GB --think_time=0.003 --detailed_view=Y --
warmup_time=1 --report_rate=1 --csv_output=<drive path>
```
9. Monitor the Turbonomic console for recommended changes and make note of recommendations.
10. Once the 60-minute testing cycle completes, collect the performance data and reset the environment, per step 2 above.
11. Implement all recommendations given by Turbonomic during the first test.
12. Begin a second test run, repeating steps 3 through 8, this time implementing any non-disruptive Turbonomic recommendations during the test run (e.g., host migrations).
13. Collect the performance data, implement any disruptive recommendations, and reset the environment.
14. Perform a third test run, implement non-disruptive Turbonomic recommendations, and collect the performance data.

	1-20	21-40	41-60	61-80	81-100
DB_SIZE	5GB	40GB	20GB	40GB	40GB
THREAD_COUNT	8	24	12	32	12
RUN_TIME	120	120	120	120	120
THINK_TIME	0.003	0.003	0.003	0.003	0.003
WARMUP_TIME	1	1	1	1	1
RAMP_RATE	100	100	100	100	100

Figure 14: DS2 configuration groups.

Detailed virtual machine configuration

Figures 15 and 16 show the detailed information of the virtual machines we used in testing before implementing Turbonomic recommendations. vDisk sizes are Thin Provisioned allocation.

VM name	Qty	OS	vCPUs	Mem (GB)	OS vDisk (GB)	DB vDisk (GB)	Log vDisk (GB)
SQL Server 2014 (Small)	40	Windows 2012 R2	2	4	40	60	80
SQL Server 2014 (Large)	60	Windows 2012 R2	2	4	40	120	40

Figure 15: Detailed information for our virtual machines.

VM storage location groups					
GB	VM	Load	Datastore	LUN	Capacity (GB)
5	1-20 (Odd)	Light	Pool3-A	2	3,200
5	1-20 (Even)	Light	Pool3-B	8	3,200
40	21-38 (Odd)	Med-High	Pool5-A	4	2,370
40	21-38 (Even)	Med-High	Pool5-B	10	2,370
40	39	Med-High	Pool6-A		
40	40	Med-High	Pool6-B		
20	41-58 (Odd)	Medium	Pool2-A	1	1,575
20	41-58 (Even)	Medium	Pool2-B	7	1,575

VM storage location groups					
GB	VM	Load	Datastore	LUN	Capacity (GB)
20	59	Medium	Pool3-A		
20	60	Medium	Pool3-B		
40	61,66,71	High	Pool1-A	0	948
40	62,67,76	High	Pool1-B	6	948
40	61-80 (Odd)	High	Pool6-A	5	3,162
40	61-80 (Even)	High	Pool6-B	11	3,162
40	81-92 (Odd)	Medium	Pool4-A	3	1,575
40	81-92 (Even)	Medium	Pool4-B	9	1,575
40	93-100 (Odd)	Medium	Pool6-A		
40	93-100 (Even)	Medium	Pool6-B		

Figure 16: VM storage location groups.

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, websites, 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.
