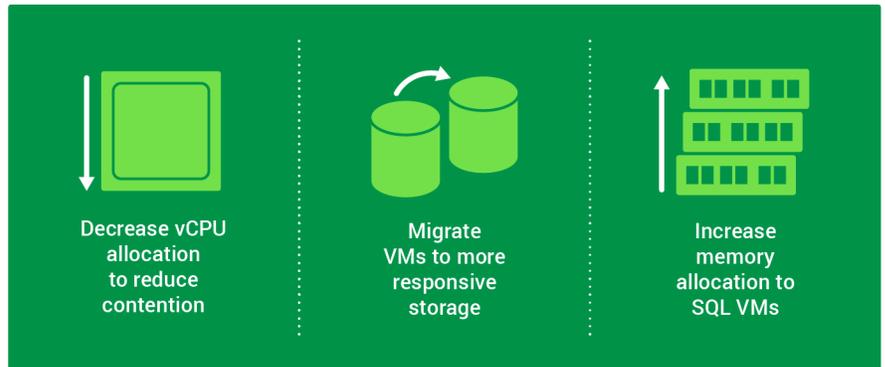


# USING TURBONOMIC TO BOOST PERFORMANCE

## Get more production from your VMs with **Turbonomic**

Turbonomic Operations Manager improved application performance in a virtual environment by recommending reallocation of vCPU, memory, and storage resources.



Up to **23%** more orders per minute\*

Up to **37%** better response times\*

\*on VMware® vSphere® 5.5 compared to a solution without Turbonomic

Assuring application performance in today's virtualized datacenters is an increasingly complex task. Out of the box, vendor-specific hypervisor tools do a good job of balancing datacenter resources for a virtual infrastructure, but do they guarantee or improve application performance? Turbonomic Operations Manager helps you make VM resource decisions at a granular level to take your application performance to the next level. Installed on a management server, Turbonomic assesses how each VM is using resources, and recommends reallocating certain resources to help you get the most performance possible from your databases.

In the Principled Technologies datacenter, we added Turbonomic to a DRS-enabled VMware vSphere 5.5 cluster to see what it could do. After running a baseline application performance test, Turbonomic recommended that we reduce vCPU count in each VM and increase memory to database VMs, enabling our databases to handle 9.6 percent more orders per minute with 17.7 percent better response times. With data from that test run, Turbonomic suggested we again increase memory allocation and move some VMs to more responsive storage, which resulted in 23.7 percent more OPM and 37.1 percent better response times compared to the baseline configuration.

In a virtualized application, every order counts. When you add Turbonomic to your environment, you can see the immediate improvement in performance from the first time you implement the recommended changes.



## WHY IS IT SO IMPORTANT TO MAXIMIZE PERFORMANCE?

Maximizing performance is vital to the success of a hosted virtual application: If you overtax your resources, users will have a degraded experience, and if you underutilize them, you'll wind up needing additional servers. That means additional expenses that include hardware costs, licensing costs, operating costs, and even datacenter space costs. By finding the right allocation for physical and virtual resources, you can optimize your environment and avoid these costly mistakes.

## HOW TURBONOMIC GETS THE JOB DONE

Turbonomic Operations Manager is a demand-driven control platform that looks at the specifics of a virtualized environment and assesses how you can make alterations that maximize resource utilization. By understanding application and VM demand and then meeting that demand with available compute, storage, and network resources, Turbonomic helps you prevent performance issues in real time. Turbonomic assures that workloads don't suffer through trial and error resource allocation when demands fluctuate. Because Turbonomic understands the demand of your applications and VMs and the available infrastructure resources at a granular level (e.g., vCPU, memory, and storage location), it assures application performance and takes it to an even higher level. Visit the Turbonomic Web site at [www.Turbonomic.com](http://www.Turbonomic.com) to learn more.

## JUST THE FACTS

We found that adding Turbonomic to our virtualized environment and following its simple recommendations increased orders per minute from 191,229 to 236,537 after two test runs. Latency, or the average response time, also decreased by 37.1 percent. (See Figure 1.)

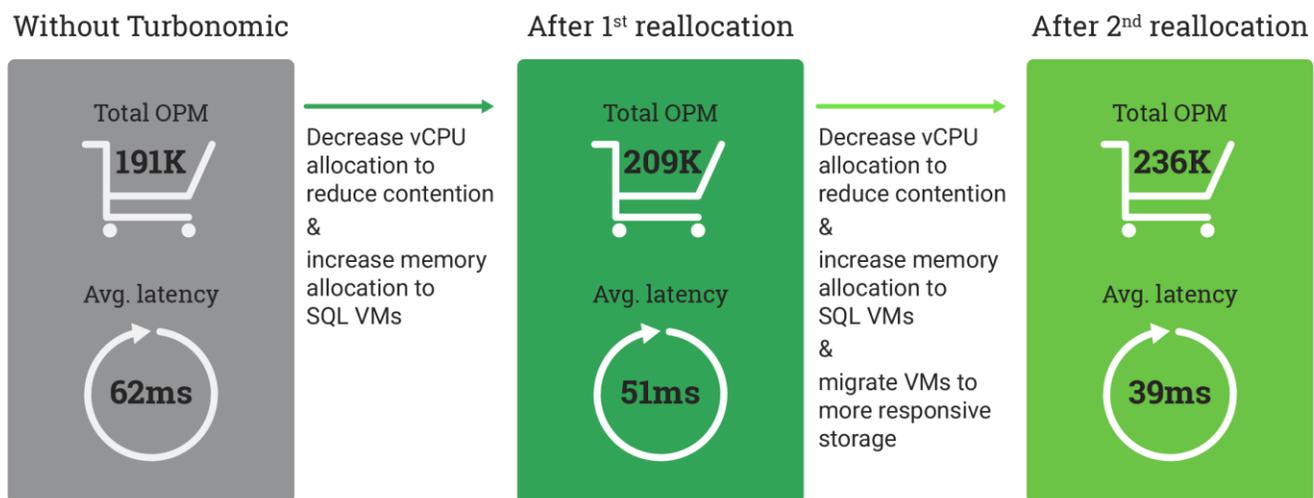


Figure 1: When we implemented Turbonomic's recommendations, application performance increased and average latency, or response time, plummeted.

## IN OUR DATACENTER

For testing, we set up a VMware vSphere 5.5 cluster on three blade servers. VMware Distributed Resource Scheduler™ (DRS) assessed resource utilization at a VM/server level, and distributed our 33 VMs equally on the servers for optimization. Twenty-one VMs ran SQL Server 2014, and 12 ran IIS Web servers for the SQL clients to connect to. We ran our ecommerce workload on 12 of the SQL Server VMs; the other 9 were idle. Our small-scale testing environment is intended to illustrate how Turbonomic works to optimize resources to boost performance in a typical three-tier application. Our testing environment used a three-server cluster to show this advantage, but the functionality of Turbonomic is not limited in any way to this scale. According to Turbonomic, the single virtual appliance we deployed in our test “scales to 10,000 virtual machines and holds two years of data.”<sup>1</sup>

See [Appendix A](#) for system configuration information and [Appendix B](#) for step-by-step testing details.

## TURBONOMIC ALLOWED US IMPROVE APPLICATION PERFORMANCE

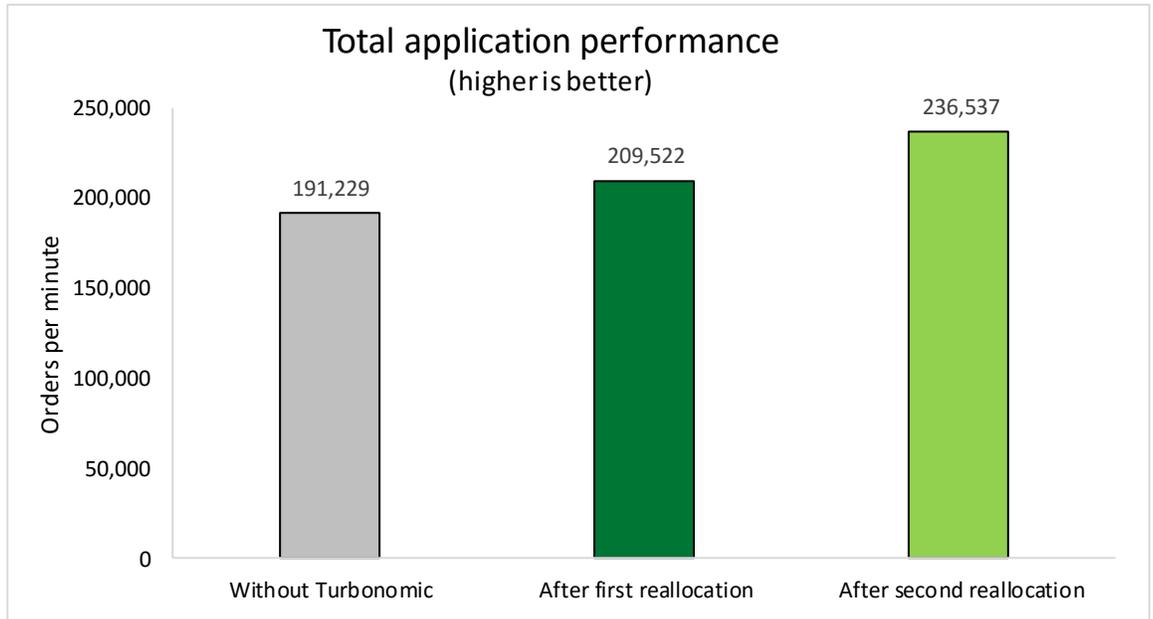
For our first test, we ran the workloads on our server cluster that had been optimized only at the VM level by VMware DRS. Turbonomic ran on a management server and gathered performance data during the baseline run. Turbonomic recommended we decrease vCPU allocation for the SQL Server VMs from 16 to 8 and allocate more memory to some SQL Server VMs to increase performance. We implemented those recommendations and ran the test again. Turbonomic’s suggestions increased application performance by 9.6 percent.

During the second run, Turbonomic continued to improve application performance with two more real-time recommendations: 1) to increase memory allocation to additional SQL Server VMs and 2) to move some VMs to more responsive storage. After we made these changes, we ran our ecommerce workload a third time. We found that performance increased even further, to 23.7 percent more than the baseline without Turbonomic recommendations. Figure 2 compares the total application performance of the servers during each test run.

---

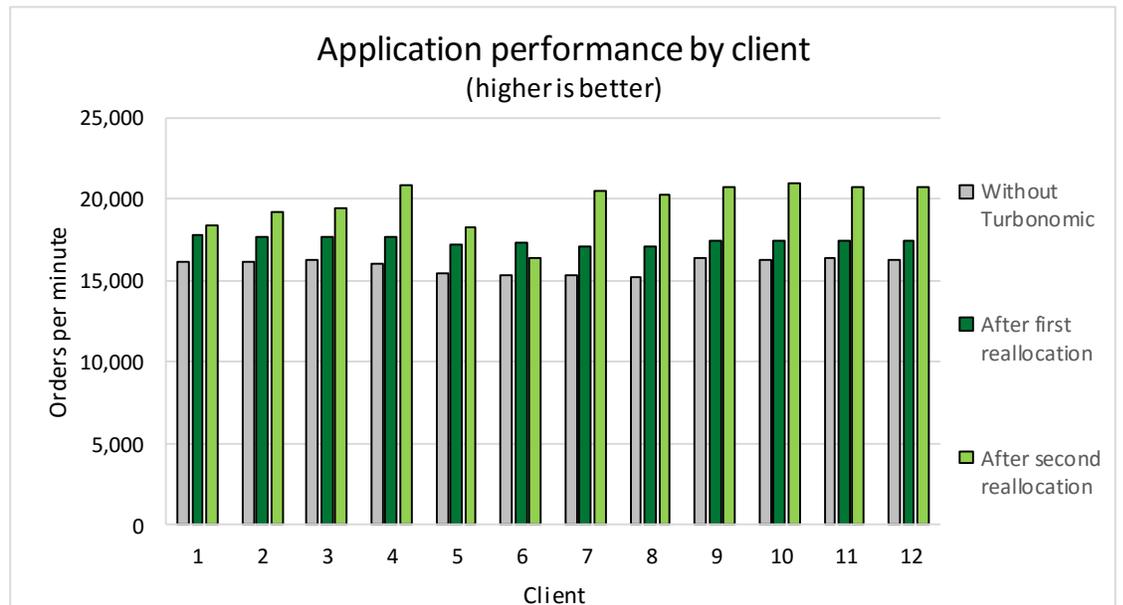
<sup>1</sup> [vmturbo.com/product/](http://vmturbo.com/product/)

**Figure 2:**  
Implementing  
Turbonomic  
recommendations  
improved total  
application  
performance.



As Figure 3 shows, though performance varied by client, implementing Turbonomic reallocation recommendations improved performance over each run.

**Figure 3:** How  
Turbonomic  
reallocation  
recommendations  
improved application  
performance, by client.



Another measure that indicates how well your applications are performing is the latency that the server experiences when completing an order. The higher the latency, the longer the wait. Figure 4 shows how the average latency dropped as we implemented Turbonomic's recommendations for better performance. By the third test run, latency decreased by 37.1 percent.

Figure 4: Average latency dropped significantly as we implemented Turbonomic's suggestions.

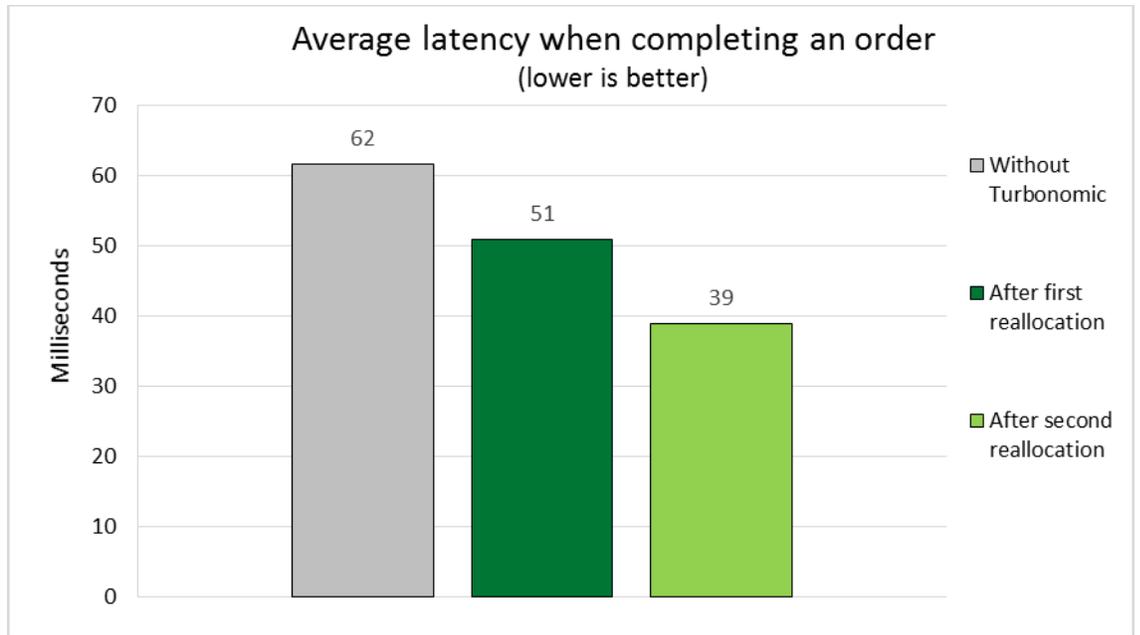
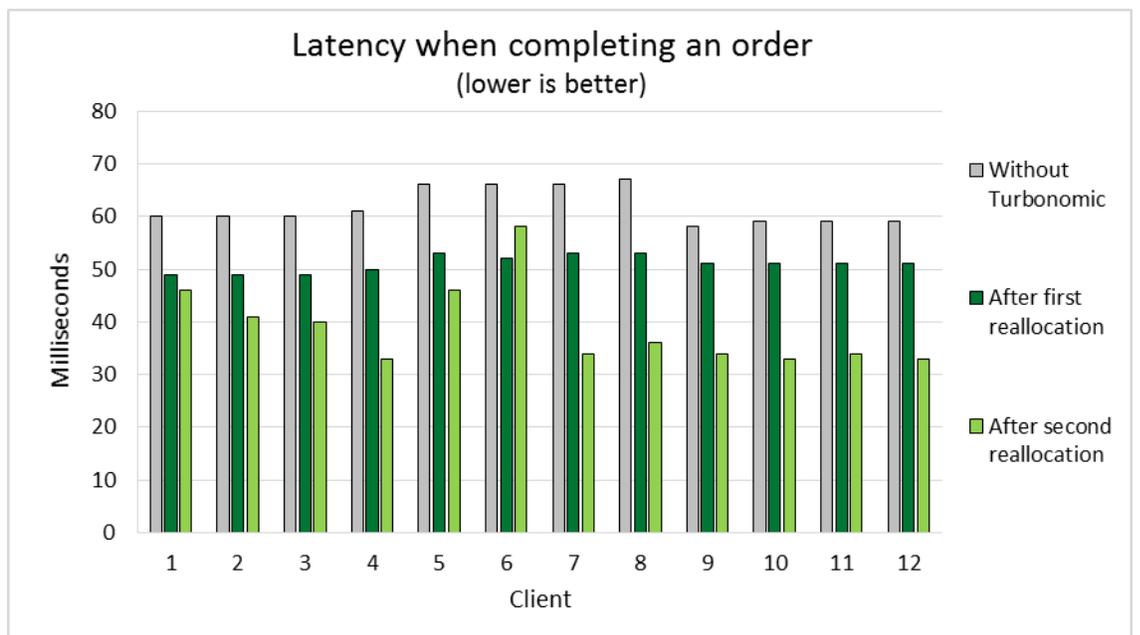


Figure 5 shows how latency decreased for each client throughout our tests.

Figure 5: Latency, in milliseconds, by client.



## CONCLUSION

Proper resource allocation is critical to achieving top application performance in a virtualized environment. Resource contention degrades performance and underutilization can lead to costly server sprawl.

We found that adding Turbonomic to a VMware vSphere 5.5 cluster and following its reallocation recommendations gave our application performance a big boost. After reducing vCPU count, increasing memory allocation to active databases, and moving VMs to more responsive storage as Turbonomic directed, online transactions increased by 23.7 percent while latency dropped significantly. Avoid the pitfalls of poorly allocated VM resources and give your virtualized application every advantage by gaining control of your environment at every level.

## APPENDIX A – SYSTEM CONFIGURATION INFORMATION

Figure 6 provides detailed configuration information for the test systems. All three Cisco UCS® B200 M2 blades were configured identically.

System	Cisco UCS B200 M2 Blade Server
<b>Enclosure</b>	
Blade enclosure	Cisco UCS 5108
<b>Power Supplies</b>	
Total number	4
Wattage of each (W)	2,500
<b>Cooling fans</b>	
Total number	8
<b>General</b>	
Number of processor packages	2
Number of cores per processor	6
Number of hardware threads per core	2
System power management policy	High Performance
<b>CPU</b>	
Vendor	Intel®
Name	Xeon®
Model number	X5760
Stepping	C0
Socket type	LGA1366
Core frequency (GHz)	2.93
Bus frequency	8.0. GT/s
L1 cache	192 KB
L2 cache	1.5 MB
L3 cache	12 MB
<b>Platform</b>	
Vendor and model number	Cisco UCS B200 M2
Motherboard model number	N20-B6625-1
BIOS name and version	Cisco® S5500.2.1.3.0.081620131102
BIOS settings	Default
<b>Memory module(s)</b>	
Total RAM in system (GB)	96
Vendor and model number	Micron® MT36KSZF1G72PZ-1G4D1AD
Type	DDR3-1333 PC3-10600
Speed (MHz)	1,333
Speed running in the system (MHz)	1,333
Size (GB)	8
Number of RAM module(s)	12
Chip organization	Double-sided

<b>System</b>	<b>Cisco UCS B200 M2 Blade Server</b>
<b>Operating system</b>	
Name	VMware vSphere 5.5.0
Build number	2143827
Language	English
<b>RAID controller</b>	
Vendor and model number	LSI Logic® SAS 1064E
Firmware version	01.32.09.00
<b>Hard drives</b>	
Vendor and model number	Seagate® ST9146803SS
Number of drives	2
Size (GB)	146
RPM	10,000
Type	SAS
<b>Network adapter</b>	
Vendor and model number	Cisco UCS M81KR

Figure 6: System configuration information for the test systems.

## APPENDIX B – HOW WE TESTED

To test Turbonomic’s ability to improve performance, we created a three-node cluster in VMware vCenter™ 5.5 and evenly distributed an OLTP-based workload across all three hosts. To simulate a realistic mixed environment of a Web-based online store, we deployed IIS-based Web servers to process the requests from the test clients, which then requested data from back-end Microsoft® SQL servers. We used DVD Store as the benchmark tool to generate the workload and evaluate the performance of the environment.

### About DVD Store Version 2.1

To create our real-world ecommerce workload, we used the DVD Store Version 2.1 (DS2) 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](http://www.delltechcenter.com/page/DVD+Store).

### Test bed configuration

Figure 7 shows a diagram of our test bed. The blade chassis was connected to a 10Gb NFS NAS storage array and an 8Gb Fibre Channel storage array.

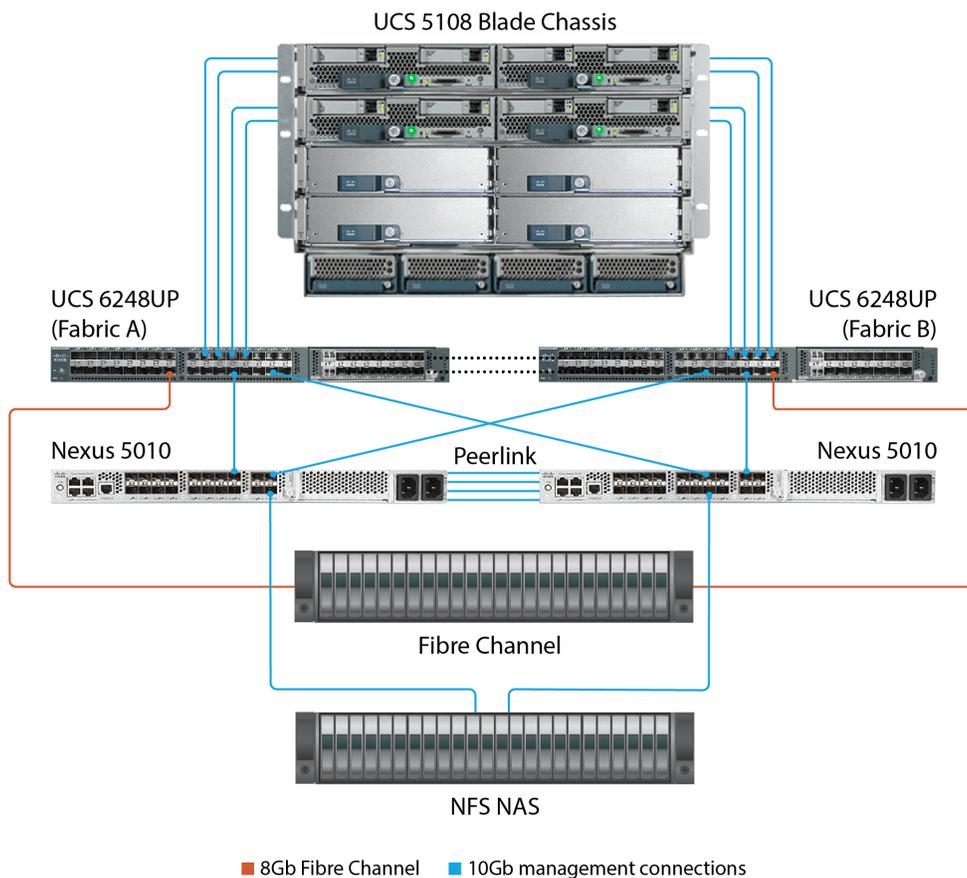


Figure 7: Test bed diagram.

## Configuring the test bed

### Installing VMware ESXi™ 5.5 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, to Accept and Continue, press F11.
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 9 in Appendix C 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 8, and click Next.
8. Choose Windows®, select Microsoft Windows Server 2012 (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 on 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 5.5

1. From the VMware vCenter 5.5 install media, click Autorun.
2. To start the install wizard, click Run.
3. Go through the Custom Install option, using the vCenter and database defined above, and install all four components (Single Sign-On, Web Client, Inventory Service, and Server).
  - a. Select the Install wizard language as English, and click OK.
  - b. At the Install wizard welcome screen, click Next.
  - c. At the End User Patent Agreement, click Next.
  - d. Agree to the License Agreement, and click Next.

- e. Enter and confirm the password you wish to use with the Administrator account for vCenter Single Sign On, and click Next.
  - f. Enter and confirm the passwords for the DBA and SQL user accounts, and click Next.
  - g. Confirm the Fully Qualified Domain Name or IP address is correct, and click Next.
  - h. Check the Use network service account checkbox, and click Next.
  - i. Accept the default installation path, and click Next.
  - j. Accept the default https port, and click Next.
  - k. Click Install.
  - l. After Single Sign On completes, and vCenter Server installation begins, enter user information and a license key, and click Next.
  - m. Select Use an existing supported database, and select the Data Source Name (DSN) for the connection to the SQL Server: `vcenter`
  - n. Type `sa` as the database username, provide the password for the SA account, and click Next.
  - o. Select the system account for the vCenter Server service account, and click Next.
  - p. Keep the vCenter default ports, and click Next.
  - q. Select 1024 MB for the JVM memory, and click Next.
  - r. To finish the vCenter server installation, click Install.
  - s. To exit the wizard, click Finish.
  - t. To confirm completion, click OK.
4. On the installer, click Exit.
  5. Restart the server.
  6. Using the vSphere client, log into the vCenter server as `TEST\administrator`
  7. Right-click the root of vCenter, and click New Data center.
  8. Name the new data center `datacenter`
  9. Add the three ESX servers under test to the data center.

### 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 4GB database size 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 4GB database in SQL Server 2012, and then performed a full backup, storing the backup file on the C: drive for quick access. We used that backup file to restore the server between test runs.

The only modification we made to the schema creation scripts were the specified file sizes for our database. 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 4GB database 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.

Figure 8 shows our initial file size modifications.

Logical name	Filegroup	Initial size (MB)
<b>Database files</b>		
primary	PRIMARY	4
cust1	DS_CUST_FG	2,048
cust2	DS_CUST_FG	2,048
ind1	DS_IND_FG	1,024
ind2	DS_IND_FG	1,024
ds_misc	DS_MISC_FG	1,024
orders1	DS_ORDERS	1,024
orders2	DS_ORDERS	1,024
<b>Log files</b>		
ds_log	Not applicable	10,240

Figure 8: Our initial file size modifications.

## 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.
4. 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 override for Move, and set the value to Automated.
2. Under Action→Host→Physical Machines by PM cluster→[Cluster name], enable the override for Suspend, and change the value to Disabled.
3. Under Discovery→Application Discovery, ensure that the MSSQL and IIS boxes are checked. For IIS, append `|w3wp.*` to the match statement. Click Apply Settings Change.
4. Under Discovery→Application Discovery→Application Credentials→Virtual Machines, override the Username and Password attributes, and enter the domain administrator authentication credentials.
5. Under Analysis→Resize Recommendation Incrementing Constants, change the Rate of Resize from Medium to High.

## Running the test

### Test start and run times

We ran all workloads concurrently 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:  

```
ds2webdriver.exe --target=<target_IP> --ramp_rate=10 --run_time=60 --  
n_threads=32 --db_size=4GB --think_time=0.05 --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.

## APPENDIX C – DETAILED VIRTUAL MACHINE CONFIGURATION

Figure 9 shows the detailed information of the virtual machines we used in testing before implementing Turbonomic recommendations.

VM name	Qty	OS	Host(s)	LUN	vCPUs	Mem (GB)	vDisk (GB)
SQL Server 2014 (Active)	4	Windows 2012 R2	Host 1	Fibre Channel	16	8	80
SQL Server 2014 (Idle)	3	Windows 2012 R2	Host 1	NFS NAS	8	8	20
IIS 8 Web Server	4	Windows 2012 R2	Host 1	NFS NAS	16	8	80
SQL Server 2014 (Active)	4	Windows 2012 R2	Host 2	Fibre Channel	16	8	80
SQL Server 2014 (Idle)	3	Windows 2012 R2	Host 2	NFS NAS	8	8	20
IIS 8 Web Server	4	Windows 2012 R2	Host 2	NFS NAS	16	8	80
SQL Server 2014 (Active)	4	Windows 2012 R2	Host 3	Fibre Channel	16	8	80
SQL Server 2014 (Idle)	3	Windows 2012 R2	Host 3	NFS NAS	8	8	20
IIS 8 Web Server	4	Windows 2012 R2	Host 3	NFS NAS	16	8	80

Figure 9: Detailed information for our virtual machines.

## ABOUT PRINCIPLED TECHNOLOGIES



Principled Technologies, Inc.  
1007 Slater Road, Suite 300  
Durham, NC, 27703  
[www.principledtechnologies.com](http://www.principledtechnologies.com)

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

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

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

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

---

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

---

**Disclaimer of Warranties; Limitation of Liability:**

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

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

---