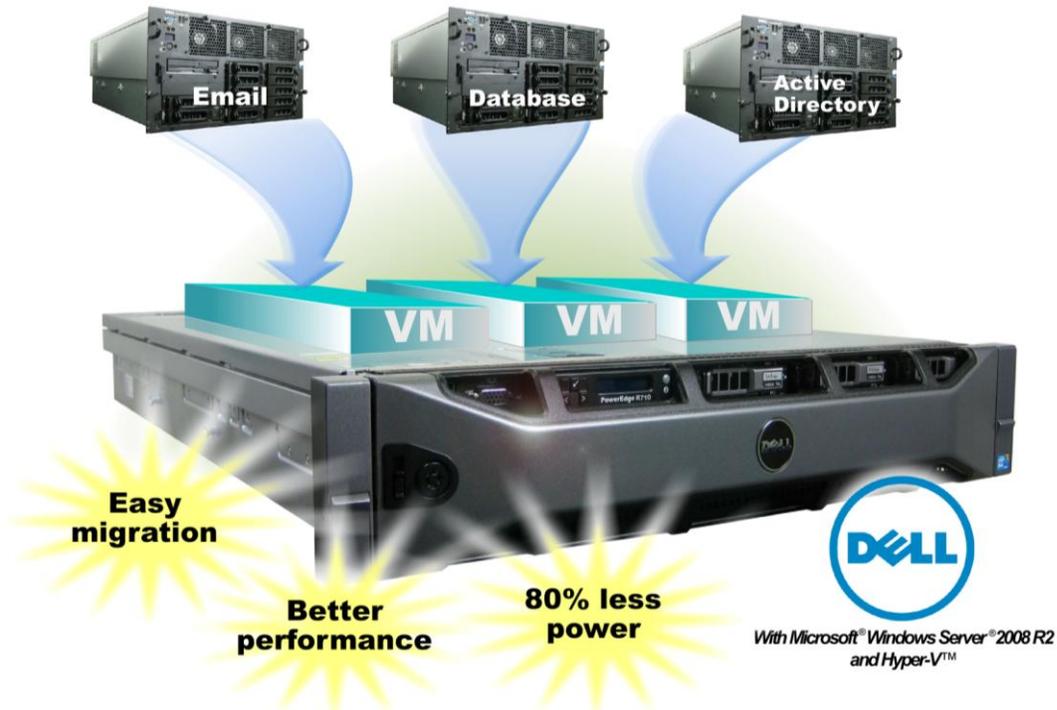


PERFORMANCE: 32-BIT TO 64-BIT SERVER TRANSITION

Replace **3** 32-bit servers with **1** Dell™ PowerEdge™ R710 64-bit server



OUR FINDINGS

In today's business environment, organizations may stay with 32-bit computing to avoid what they perceive to be the risks of upgrading their applications to 64-bit platforms. As this Principled Technologies (PT) study and the related migration guide (<http://www.principledtechnologies.com/clients/reports/Dell/32to64migration0410.pdf>) show, however, you can make the move to a 64-bit Dell PowerEdge R710 server today without changing your applications, allowing you to decommission outdated hardware; reap cost, performance, and space benefits; and use up to 80 percent less power.

OUR PROCESS

In Principled Technologies' tests in our labs, we used Microsoft virtualization technologies to consolidate three legacy servers, running Microsoft Active Directory®, Microsoft SQL Server®, and Microsoft Exchange®, to virtual machines on a single Dell PowerEdge R710. We first measured the performance of each workload on the older 32-bit servers and then performed a straightforward physical-to-virtual (P2V) migration of all three of them to one Dell PowerEdge R710 running 64-bit Microsoft Windows Server 2008 R2. We verified that each virtual machine performed better than its physical counterpart.



PROJECT OVERVIEW

We tested the following legacy servers:

- Dell PowerEdge 6800 server with two single-core processors (Active Directory)
- Dell PowerEdge 6800 server with two dual-core processors (SQL Server 2005)
- Dell PowerEdge 6800 server with two dual-core processors (Exchange 2003)

We performed a P2V migration on the above servers, transitioning them to virtual machines (VMs) on the new server. We then tested the three VMs simultaneously with the applicable workloads. For our virtualization host machine, we used a Dell PowerEdge R710 server with two quad-core processors.

For testing, all Dell PowerEdge 6800 servers ran Microsoft Windows Server 2003 R2 with SP2 Enterprise Edition. On the server we designated for Active Directory, we ran DCPromo to promote it to a domain controller for our test bed. On the server we designated for SQL Server, we installed Microsoft SQL Server 2005 with SP3. On the server we designated for Microsoft Exchange, we installed Microsoft Exchange Server 2003 with SP2. The virtualization host server ran Microsoft Windows Server 2008 R2 Enterprise Edition with the Hyper-V role enabled. This report shows the performance and energy-savings gains you can realize by migrating from three legacy 32-bit servers to a single 64-bit server. For more details on how to virtualize your environment through the P2V process, see our accompanying how-to Migration Guide at <http://www.principledtechnologies.com/clients/reports/Dell/32to64migration0410.pdf>.

WORKLOAD

Each of our three legacy servers had a specific application role to simulate a typical business environment. Below, we discuss these three application roles—Active Directory, database, and email—and the workload we chose to test each.

Active Directory

We used the default settings with the ADTest tool to stress the Active Directory server in our test bed. ADTest is a load-generation tool that allows the simulation of client transactions on a host server, performing a wide range of login functions, with the options of other Active Directory activities. ADTest performs the instructed pre-configured test in an endless loop and reports operations per second for the main metric of the Interactive Logon test. The pre-configured Interactive Logon test we used for our testing in ADTest includes the following user simulations:

- Logon

- CrackNameGuid
- CrackNameDN
- GetMachineTicket
- Bind
- NamingContextSearch
- PolicySearch
- FirstSiteSearch

We ran the simulation for 30 minutes against the Dell PowerEdge 6800 and recorded the operations per second, averaged over the course of the complete test run. After migrating to the VM running Active Directory on the Dell PowerEdge R710, we performed an identical test, simultaneously running with the remaining two workloads.

For more details on the ADTest tool, see

<http://www.microsoft.com/downloads/details.aspx?FamilyID=4814fe3f-92ce-4871-b8a4-99f98b3f4338&displaylang=en>.

Database

To build the SQL Server workload, we used DVD Store Version 2 (DS2), an open-source simulation of an online e-commerce DVD store. DS2 has database components and Web server components, and includes driver programs that put heavy loads on these components. We used the included driver program to stress the database component. In our tests, we used one Microsoft SQL Server 2005 instance, and we configured this SQL Server 2005 instance with a 5GB database.

The main DS2 metric is orders per minute (OPM), which the driver program calculates and reports via the Windows Performance Monitor utility on the client machine. We used the output from the driver program to record OPM, and we report the last OPM score the benchmark reported.

When DS2 is executing, simulated customers log in; browse movies by actor, title, or category; and purchase movies. Each DS2 order consists of a customer login, a number of searches for movies, and a purchase. The searching function uses title, actor, or category as a parameter. The title and actor searches use full-text search. The other customer actions, including adding new customers, exercise a wide range of database functions.

We did not use the front-end Web client component of DS2, but instead ran the included driver application on a client machine directly via its command-line interface. We used the default DS2 parameters and setup configuration, with the exceptions we note in the How we tested section of this report.

The client machine ran a single instance of DS2, with 30 threads, and used 0.1 seconds think time. This simulated a 300-user environment, with each user executing approximately one operation per second.

The DS2 driver application creates an OPM performance counter on the client. While the DVD Store client application outputs OPM at 10-second intervals visually, we chose to collect this OPM metric via a performance monitor counter on each client at 1-second intervals. We ran this workload on the legacy database server for 30 minutes, and then performed an identical test on the database VM on the Dell PowerEdge R710 while simultaneously running the virtualized Active Directory and email workloads for 30 minutes.

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

Email

We use Microsoft Exchange Server 2003 Load Simulator (LoadSim) to measure email performance. LoadSim uses the MAPI Messaging Benchmark 3 (MMB-3) to simulate MAPI email clients. The simulated clients issue requests to an Exchange server to generate an email load. LoadSim measures both the server's response time and the number of messages it delivers. LoadSim is available for free download from <http://www.microsoft.com/downloads/details.aspx?FamilyId=92EB2EDC-3433-47CA-A5F8-0483C7DDEA85&displaylang=en>.

LoadSim results show the response time each for each server, the time in milliseconds it took to complete a request. Because the object of the server is to handle as many requests as possible, lower scores, which show a server able to handle more requests in a given time, are better. Generally, the most important point is the median 95th percentile, which means 95 percent of the time the server handled request at that level. We report the latencies in milliseconds (ms) at this median 95th percentile.

We created a custom workload and adjusted it for 300 users that LoadSim distributed across a single storage group. We ran the benchmark for 30 minutes on the legacy email server, and then performed an identical test on the Exchange VM on the Dell PowerEdge R710 server while simultaneously running the virtualized Active Directory and database workloads. For the client machine, we used a server running Windows Server 2003 R2 Enterprise Edition.

SYSTEM COMPARISON

Figure 1 shows a side-by-side comparison of the key hardware differences between the four physical servers. For detailed system configuration information, see Appendix A.

Hardware specifications	Dell PowerEdge 6800 (Active Directory server)	Dell PowerEdge 6800 (database server)	Dell PowerEdge 6800 (email server)	Dell PowerEdge R710 (virtualization server)
CPU	Intel Xeon Processor MP	Intel Xeon Processor MP 7020	Intel Xeon Processor MP 7020	Intel Xeon Processor X5570
CPU speed (GHz)	3.16	2.66	2.66	2.93
Number of processor packages	2	2	2	2
Number of cores per processor package	1	2	2	4
Number of hardware threads per core	2	2	2	2
Total number of logical processors	4	8	8	16
Hard drive	2 x 73GB 10K SCSI	5 x 73GB 10K SCSI	5 x 73GB 10K SCSI	8 x 146GB 15K SAS
Memory type	PC2-3200R ECC DDR2	PC2-3200R ECC DDR2	PC2-3200R ECC DDR2	PC3-10600R DDR3
Total memory (GB)	4	4	4	24

Figure 1: Key hardware specifications for the four physical servers.

The Dell PowerEdge 6800 that ran Active Directory contained two hard drives, while the remaining two Dell PowerEdge 6800 servers each contained five hard drives. For all three Dell PowerEdge 6800 servers, we configured a mirrored (RAID 1) volume on which to install the operating system and application software. We then configured a 25GB partition on each system volume. On the database and email servers, we configured the remaining three disks in a RAID 5 configuration (striped with parity), and created a 40GB partition on those volumes to hold SQL Server data and Exchange data, respectively.

On the Dell PowerEdge R710, we configured a two disk mirrored volume (RAID 1), on which we installed the host operating system, Windows Server 2008 R2 Enterprise Edition. Additionally, we stored each

VM's operating system virtual hard disk (VHD) file on this volume. We then created two additional logical disks, each a three-disk RAID 5 volume. We designated one of these volumes for the migrated SQL Server 2005 data VHD, and the other for the migrated Exchange 2003 data VHD. Because of the high input/output (I/O) requirements of both SQL Server 2005 and Exchange 2003, we wanted to isolate each application's physical disk assignments. This also closely mimicked the environment of the original legacy systems, which included a three-disk RAID 5 volume for both SQL Server and Exchange data.

Figure 2 shows details of our virtual machine configuration on the Dell PowerEdge R710 after P2V conversion.

Virtual machine specifications	Active Directory VM	SQL Server 2005 VM	Exchange 2003 VM
Virtual processors	4	4	4
RAM (GB)	4	4	10
Operating system VHD location on host	C: (2 disk RAID 1)	C: (2 disk RAID 1)	C: (2 disk RAID 1)
Application data VHD location on host	N/A	D: (3 disk RAID 5)	E: (3 disk RAID 5)
Operating system VHD disk type	Dynamic	Dynamic	Dynamic
Application data VHD disk type	N/A	Dynamic	Fixed
Virtual storage controller used	IDE	IDE	IDE

Figure 2: Virtual machine configuration for the three virtual machines under test on the Dell PowerEdge R710.

We used Microsoft System Center Virtual Machine Manager 2008 R2 (SCVMM) to perform a P2V conversion on each of the Dell PowerEdge 6800 legacy servers, transitioning them to a virtual environment, powered by Hyper-V, on the Dell PowerEdge R710. As we note in Figure 2 above, we left RAM allotments identical except in the case of the SQL Server virtual machine, where we allotted 10 GB of RAM to the SQL Server virtual machine and then configured the VM to use Physical Address Extensions (PAE) and the SQL Server instance inside that VM to use AWE (Address Windows Extensions). For details on the steps we followed, see the How we tested section. As we also note in Figure 2 above, we isolated SQL Server disk I/O and Exchange disk I/O at the host level, assigning each virtual machine a VHD for holding that specific applications' data.

For all P2V conversion, we accepted the default options, which specified the use of the virtual IDE controller and the use of dynamically sized VHD files. For the Exchange data VHD, we used the VHD conversion feature in Hyper-V after the P2V process to convert the VHD assigned to hold Exchange data to a fixed-size

format to avoid any file growth during the test period. For more details on the steps we followed, see the How we tested section.

WHAT WE FOUND

The three applications performed faster and better virtualized on the 64-bit host than they did on standalone physical legacy servers, even while the 64-bit host ran all three workloads simultaneously. We also found that the transition to a single 64-bit Microsoft Windows Server 2008 R2 platform with Hyper-V yielded a power savings of 80 percent. Below, we discuss each application's gain in performance and show our power measurements.

Figure 3 compares the operations-per-second results from the ADTest workload for the physical server running Active Directory and the Active Directory VM. The Dell PowerEdge 6800 delivered only 120 operations per second, while the virtual

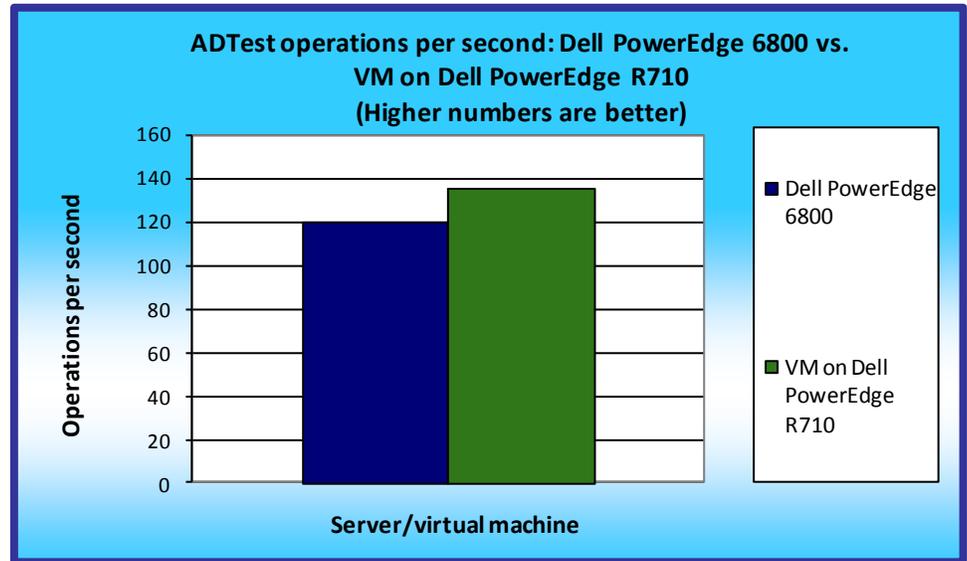


Figure 3: ADTest results, in operations per second, for the Dell PowerEdge 6800 and the Active Directory VM running on the Dell PowerEdge R710. Higher numbers are better.

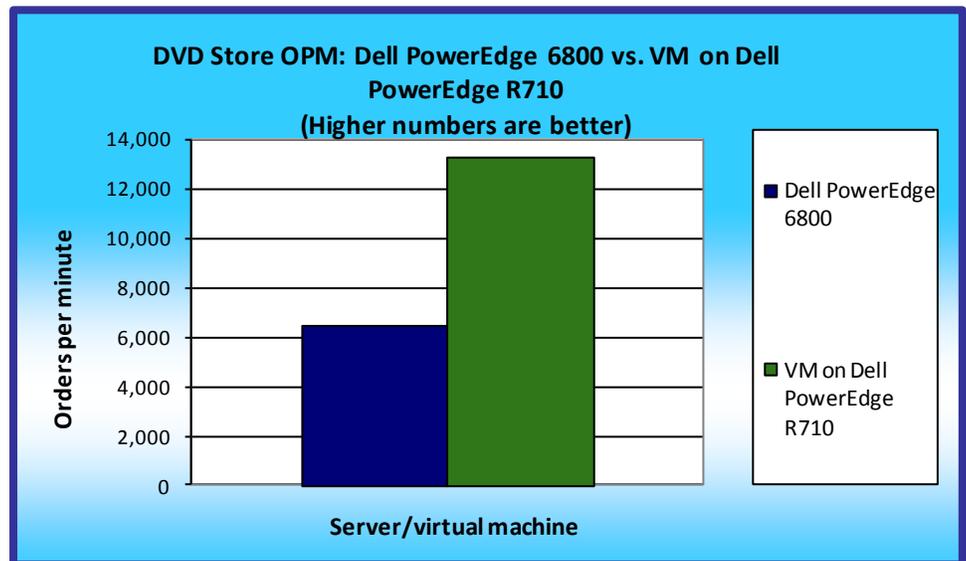


Figure 4: DVD Store Version 2 performance results, in OPM, for the Dell PowerEdge 6800 and the SQL Server 2005 VM running on the Dell PowerEdge R710. Higher numbers are better.

machine running Active Directory on the Dell PowerEdge R710 achieved 135 operations per second, a 13 percent performance increase for the VM on the Dell PowerEdge R710.

Figure 4 compares the orders-per-minute results of the DS2 test on the Dell PowerEdge 6800 running SQL Server 2005 and the VM on

the Dell PowerEdge R710 running SQL Server 2005. The SQL Server virtual machine on the Dell PowerEdge R710 yielded a DVD Store score of 13,313 orders per minute (OPM), while the Dell PowerEdge 6800 achieved only 6,515 OPM, a 104 percent performance increase for the VM on the Dell PowerEdge R710.

Figure 5 compares the latency results of the LoadSim test on the physical Exchange 2003 server and the virtualized Exchange 2003 server. Based on the 95th percentile median score, the VM running Exchange 2003 on the Dell PowerEdge R710 delivered results 5 percent faster than the Dell PowerEdge 6800 running Exchange 2003. The Exchange 2003 latency on the Dell PowerEdge R710 VM was 174 milliseconds (ms), while the latency on the Dell PowerEdge 6800 was 183ms.

Figure 6 summarizes the scores of the three workloads on the two platforms.

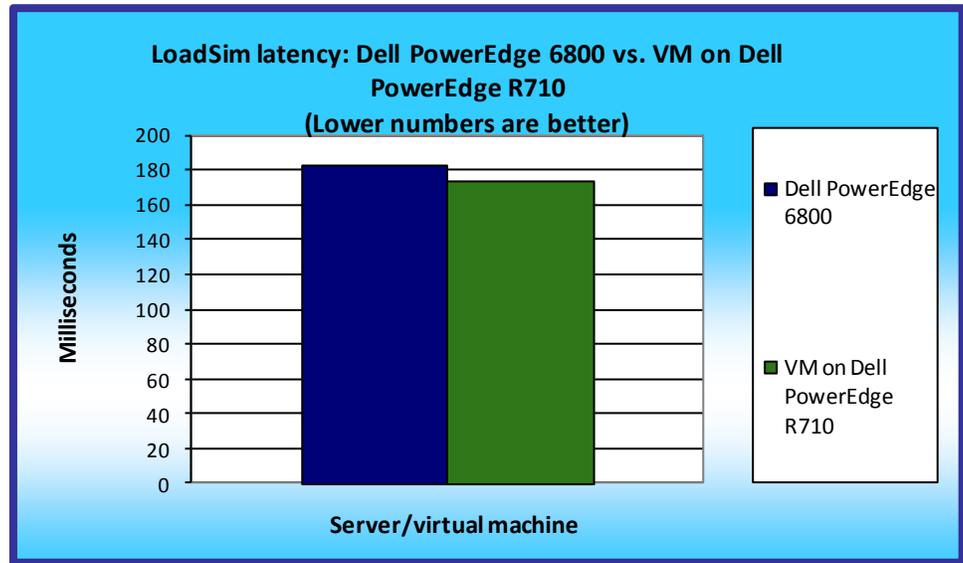


Figure 5: LoadSim 2003 latency results, in ms, for the Dell PowerEdge 6800 and the Exchange 2003 VM running on the Dell PowerEdge R710. Lower numbers are better.

Workload	32-bit physical legacy servers	64-bit Dell PowerEdge R710 running application-specific VMs	Percentage improvement
ADTest: operations per second (higher numbers are better)	120	135	13%
DVD Store: orders per minute (higher numbers are better)	6,515	13,313	104%
LoadSim 2003: latency in ms (95th Percentile) (lower numbers are better)	183	174	5%

Figure 6: Median results from each test on the two platforms.

Figures 7 and 8 show the dramatic power savings achieved by consolidating the legacy environment consisting of three Dell PowerEdge 6800 servers to the new virtualized environment on one Dell PowerEdge R710. The Dell PowerEdge R710 running VMs used 80 percent less power than the three Dell PowerEdge 6800 legacy servers.

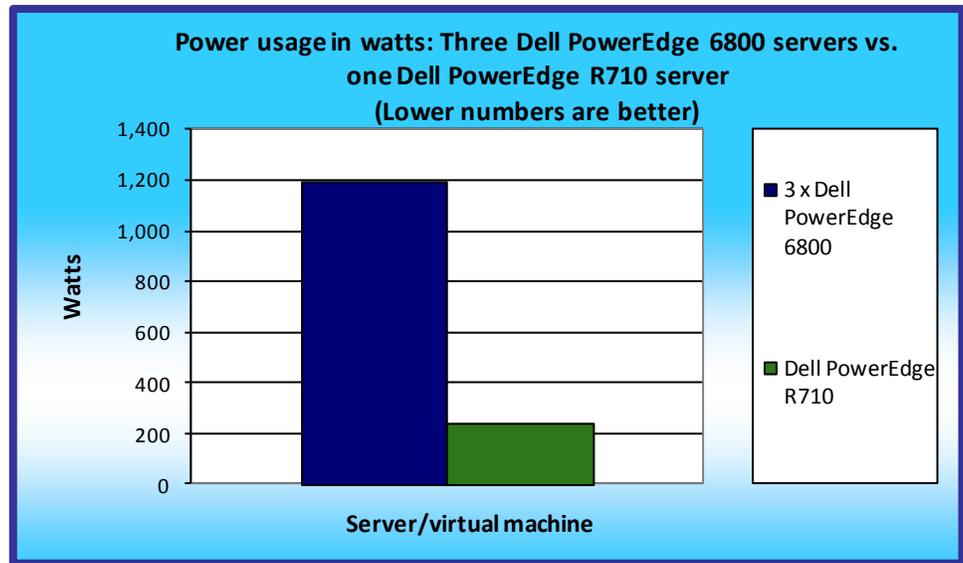


Figure 7: Power usage, in watts, for the three Dell PowerEdge 6800 servers and the one Dell PowerEdge R710. Lower numbers are better.

Server	Power usage in watts
Dell PowerEdge 6800 – Active Directory	406.7
Dell PowerEdge 6800 – SQL Server 2005	423.2
Dell PowerEdge 6800 – Exchange Server 2003	363.5
Three Dell PowerEdge 6800 servers - Total	1,193.4
Dell PowerEdge R710 –Active Directory VM, SQL Server 2005 VM, and Exchange Server 2003 VM	243.8

Figure 8: Power usage, in watts, for the three Dell PowerEdge 6800 servers and the one Dell PowerEdge R710. Lower numbers are better.

HOW WE TESTED

Setting up the Dell PowerEdge 6800 servers

We installed Windows Server 2003 R2 Enterprise Edition with SP2 on each Dell PowerEdge 6800 server, and then installed the applications necessary to run each workload on the applicable server: Active Directory, Microsoft SQL Server 2005, and Microsoft Exchange Server 2003.

Configuring the storage on each Dell PowerEdge 6800

1. On each server, create a logical disk containing two physical disks, and assign a RAID 1 policy.
2. On the Exchange and SQL Server machines, create an additional logical disk containing three physical disks, and assign a RAID 5 policy.

Installing Windows Server 2003 R2 Enterprise Edition on each Dell PowerEdge 6800

1. Insert the first installation CD into the drive, and boot the system.
2. At the Welcome to Setup screen, press Enter.
3. At the Windows Licensing Agreement screen, press F8 to agree to the terms.
4. Select the appropriate unpartitioned space, and press C to create a partition. (We chose to create a 25GB partition.)
5. Select Format the partition using the NTFS file system, and press Enter.
6. At the Regional and Language Options screen, click Next.
7. Enter Name and Organization, and click Next.
8. At the Your Product Key screen, enter your product key, and click Next.
9. At the Licensing Modes screen, click Next.
10. At the Computer Name and Administrator Password screen, choose an appropriate name and password, and click Next.
11. At the Date and Time Settings screen, change the Date & Time and Time Zone if appropriate, and click Next.
12. After the system restarts automatically, follow the prompt to log into the server.
13. At the Windows Setup screen, insert the second installation CD when prompted, and click OK.
14. At the Windows Server 2003 R2 Setup Wizard Welcome screen, click Next.
15. At the Windows Server 2003 R2 Setup Wizard Summary screen, click Next.
16. Click Finish.
17. Assign a static IP address to each Dell PowerEdge 6800 in the same subnet.
18. On each server, using the Notepad text editor, add the /3GB switch to the boot.ini file
19. Reboot the servers.

Note: We repeated the install for each Dell PowerEdge 6800 server and applied all current Windows updates to each machine.

Installing Active Directory on the domain controller server

1. Select Start→Administrative Tools→Manage Your Server.
2. Click Add or remove a role.
3. At the Configure Your Server window, click Next.
4. At the Configuration and Options screen, select Custom Configuration, and click Next.
5. At the Server Role screen, select Domain Controller, and click Next.
6. At the Summary of Selections screen, click Next.
7. At the Welcome pop-up window, click Next.
8. At the Operating System Compatibility screen, click Next.
9. For the Domain Controller Type, leave the default option (Domain controller for a new domain), and click Next.
10. At the Create New Domain screen, leave the default option (Domain in a new forest), and click Next.
11. At the Install or Configure DNS window, select No, and click Next.
12. At the New Domain Name screen, type a domain name, and click Next.
13. At the NetBIOS Domain Name screen, enter a NetBIOS domain name, and click Next.

14. Click Next through the following screens: Database and Log Folders, Shared System Volume, and Permissions. Doing so accepts the default options for each of those screens.
15. At the Directory Services Restore Mode Administrative Password screen, enter a password, and click Next.
16. When the installation software prompts you to do so, insert the operating system CD, and click OK.
17. At the Completing the Active Directory Installation Wizard screen, click Finish.
18. When the system prompts you to reboot the server, click Restart Now.
19. After the server reboots, log into the system.
20. At the This Server is Now a Domain Controller window, click Finish.

Installing Microsoft SQL Server 2005 on the database server

1. Insert the SQL Server 2005 installation media into the server.
2. Navigate to the SQL Server 2005 DVD, and double-click. Choose the appropriate edition if the installer prompts you with a menu.
3. Under the Install section, click the Server components link to begin the installation.
4. At the End User License Agreement, accept the licensing terms, and click Next.
5. After installing the prerequisites, click Next.
6. At the Welcome screen, click Next.
7. At the System Configuration Check, click Next. For this installation, you can ignore the possible warning regarding IIS, as it is unnecessary for this configuration.
8. At the Registration Screen, enter the appropriate name and company.
9. At the Components to Install screen, select SQL Server Database Services and Workstation components.
10. Click Advanced.
11. Uncheck Business Intelligent Development Studio, uncheck Replication, and uncheck Books Online & Sample Databases.
12. Click Next to start the installation.
13. At the Instance Name screen, click the Default instance radio button, and click Next.
14. At the Service Account screen, select the Use the built-in System account radio button.
15. Select Local system, and click Next.
16. At the Authentication Mode screen, select Mixed Mode.
17. Specify the sa logon password by entering a password in both fields, and click Next.
18. At the Collation Settings screen, click Next.
19. Accept the default Error and Usage Report Settings, and click Next.
20. At the Ready to Install screen, click Install to begin the installation.
21. At the end of the installation in the Setup Progress screen, click Next.
22. Click Finish to complete the installation.
23. Click Start → All Programs → Microsoft SQL Server → SQL Server Management Studio. Connect to the SQL Server instance.
24. Create a SQL Server login for the ds2user.
25. Reboot the server before installing SQL Server SP3.

Installing Service Pack 3 for Microsoft SQL Server 2005

SQL Server 2005 Service Pack 3 is available for download from

<http://www.microsoft.com/downloads/details.aspx?FamilyID=ae7387c3-348c-4faa-8ae5-949dfbe59c4&displaylang=en>

1. Run SQLServer2005SP3-KB955706-x86-ENU.exe.
2. At the Welcome screen, click Next.
3. At Accept the License Agreement, click Next.
4. At the Feature Selection screen, keep the defaults, and click Next.
5. At the Authentication screen, click Next.
6. At the Error and Usage Reporting Settings screen, click Next.
7. At the Running Processes screen, click Next.
8. Click Install.
9. If a compatibility error occurs during the installation, click Run program.
10. Click Next.
11. At the Installation Complete screen, click Next.
12. Click Finish.
13. Reboot the server.

Installing Microsoft Exchange Server 2003 on the mail server

1. Install Internet Information Services 6.0 on the mail server.
 - a. Select Start→Control Panel→Add or Remove Programs.
 - b. Click Add/Remove Windows Components.
 - c. Select Application Servers, and click Details.
 - d. Click ASP.NET, and make sure a check appears in the check box.
 - e. Select Internet Information Services (IIS), and click Details.
 - f. Click NNTP Services and SMTP Services, make sure checks appear in both check boxes, and click OK.
 - g. Click OK to close the Application Server window.
 - h. At the Windows Components Wizard, click Next to begin the installation.
 - i. When the system prompts you to do so, insert the OS CD, and click OK.
 - j. At the Completing the Windows Components Wizard window, click Finish.
 - k. Close the Add or Remove Programs window.
2. Insert the Microsoft Exchange Server 2003 CD. The CD should automatically launch the installation software.
3. Select Exchange Server Enterprise Edition.
4. Run the Setup application.
5. Click the Exchange Deployment Tools link.
6. Click the Deploy the first Exchange 2003 Server link.
7. Click the New Exchange 2003 Installation link.
8. Click the Run ForestPrep now link.
 - a. When the compatibility issue notice appears, click Continue, and check Don't display this message again.
 - b. At the Welcome screen, click Next.

- c. Accept the End User License Agreement, and click Next.
 - d. At Component Selection Screen, keep the defaults, and click Next.
 - e. Leave the default account information, and click Next.
 - f. Installation should start. (This process may take a while to complete.)
 - g. When the installation completes, click Finish.
9. Return to the Exchange Server Deployment Tools screen, and click the Run DomainPrep now link.
 - a. At the Welcome screen, click Next.
 - b. Accept the End User License Agreement, and click Next.
 - c. At Component Selection Screen, keep the defaults, and click Next.
 - d. At the insecure domain notice, click OK.
 - e. When the installation completes, click Finish.
 10. Return to the Exchange Server Deployment Tools screen, and click Run Setup Now.
 - a. At the Welcome screen, click Next.
 - b. Accept the End User License Agreement, and click Next.
 - c. At the Component Selection screen, the default selection for Microsoft Exchange should be Typical. If it is not, use the drop-down box to select Typical.
 - d. At this point, enter the storage location for your mail database. In our case, we used drive E:\ to store our Exchange installation and data.
 - e. Click Next.
 - f. For the Installation Type, select Create new Exchange Organization, and click Next.
 - g. Enter *Domain Name* as the Organization Name, and click Next.
 - h. Accept the Microsoft Exchange Server Licensing Agreement, and click Next.
 - i. Review the installation summary, and click Next.
 - j. The installation should start. (This process may take a while to complete.)
 - k. When the installation completes, click Finish.
 11. Close the Exchange Server Deployment Tools window.
 12. Install Microsoft Exchange Server 2003 Service Pack 2.
 - a. Insert the SP2 CD, and double-click setup.exe.
 - b. Accept the default location to extract the file, and click OK.
 - c. After the extraction finishes, open Windows Explorer, and browse to the update.exe executable.
 - d. At the Welcome screen, click Next.
 - e. Select I agree to accept the License Agreement, and click Next.
 - f. At the Component Selection window, leave the default option (Update), and click Next.
 - g. Verify the installation Summary, and click Next.
 - h. When the installation completes, click Finish.
 13. Select Start→All Programs→Microsoft Exchange→System Manager.
 14. In the left pane of the window, expand Servers and MailServer so the First Storage Group is visible.
 15. Right-click First Storage Group, and select Properties.
 16. Check the Enable circular logging check box.
 17. Click Apply, and click Okay to return to Exchange System Manager screen.
 18. Under First Storage Group, right-click Mailbox Store, and select Properties.
 19. Select the Database tab.
 20. Next to Maintenance interval, click Customize.
 21. Remove all blue from the boxes so the system will not perform maintenance, and click OK.

22. Click OK to close Mailbox Store Properties window.

Setting up the host virtualization server

We used the instructions we provide in our companion how-to Migration Guide to install Windows Server 2008 R2, enable the Hyper-V role, install SCVMM, and perform a P2V migration of the three legacy servers to virtual machines on one Dell PowerEdge R710. For more information, see the Guide at <http://www.principledtechnologies.com/clients/reports/Dell/32to64migration0410.pdf>. After the P2V conversion, the only changes we made to the virtual machine configurations were as follows:

1. Convert the VHD containing Microsoft Exchange data, on the Microsoft Exchange VM, from a dynamically sized VHD file to a fixed-size VHD file:
 - a. In Hyper-V Manager, launch the Edit Virtual Hard Disk Wizard.
 - b. Specify the VHD file, and choose Convert.
 - c. Follow the wizard default options to convert the VHD.
2. Modify the memory configuration on the VM running SQL Server 2005:
 - a. In Hyper-V Manager, allocate 10GB RAM to the VM:
 - i. Open Hyper-V Manager.
 - ii. Ensure the VM is not running.
 - iii. Right-click the VM, and choose Settings.
 - iv. Click Memory, and specify 10240MB of RAM.
 - v. Click OK to apply the changes.
 - b. Enable PAE in Windows Server 2003 on the VM, and then enable AWE for SQL Server 2005:
 - i. In Hyper-V Manager, select the VM running SQL Server 2005, and choose Action, Start.
 - ii. In Hyper-V Manager, right-click the VM running SQL Server 2005, and choose Connect.
 - iii. Log into the VM.
 - iv. In the VM, click Start → Run, and type the following:

```
notepad c:\boot.ini
```
 - v. Add the following switch to the end of the boot parameters:

```
/PAE
```
 - vi. Reboot the VM.
 - vii. After the VM restarts, log into the VM.
 - viii. Click Start, All Programs, Microsoft SQL Server 2005, SQL Server Management Studio.
 - ix. Connect to the SQL Server instance.
 - x. Click New Query.
 - xi. Enter the following T-SQL in the query window to configure the SQL Server instance with AWE usage of 9GB:

```
sp_configure 'show advanced options', 1  
RECONFIGURE
```

```
GO
```

```
sp_configure 'awe enabled', 1  
RECONFIGURE  
GO
```

```
sp_configure 'min server memory', 1024  
RECONFIGURE  
GO
```

```
sp_configure 'max server memory', 9216  
RECONFIGURE  
GO
```

- xii. Press F5 to execute the query.
- xiii. Restart the SQL Server service by right-clicking the server in Object Explorer and choosing Restart.

Configuring the Active Directory workload

We downloaded the ADTest tool, which is part of the Windows Server 2003 Resource Kit, from Microsoft. We used the following steps to install ADTest and configure the Active Directory with the basic necessary structure needed for testing. ADTest installs detailed configuration and setup documentation in the install directory. We installed the ADTest client component on two separate clients and executed both clients against the server simultaneously.

Installing ADTest on the sever

1. Copy the adtest.msi file to the server, and double-click to execute.
2. Proceed through the installation, accepting all default options.
3. As instructed in the ADTest documentation, change User Rights in the Group Policy editor. You must modify the following: Log on as Batch, Act as part of the operating system, Log on as Service, and Replace process level token.
 - a. Click Start→Run, and type MMC to launch Microsoft Management Console (MMC).
 - b. Under Console, click Add/Remove Snap-in, select Group Policy Snap-in, and click Add.
 - c. Click Browse, and select Default Domain Policy, and click Finish.
 - d. Repeat steps b and c, but select DomainControllers.domain name.Local and then Default Domain Controller Policy. Click Finish.
 - e. Navigate to Computer Configuration→Windows Settings→Security Settings→Local Policies→User Rights Assignment, and add the Everyone group to the policies we mention above. You should perform this on the domain and on the domain controller you added in steps b through d.
4. Modify the RID Pool Block Size in the registry:
 - a. Click Start→Run, and type Regedit

- b. Navigate to `\HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\NTDS\RID Values\RID Block Size`, and change the setting to `REG_DWORD 0x8000`.
5. Change TCP/IP setting in the registry:
 - a. Click Start→Run, and type `Regedit`
 - b. Navigate to `\HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Tcpip\Parameters\` and add `TCP WindowSize` with a value of `REG_DWORD 0xffff`
 - c. Add new key `MaxUserPort` with a value of `REG_DWORD 0xffffe`

Setting up the ADTest Clients

1. Install Windows Server 2003 as we did with the legacy servers from above with latest updates, and join the computer to the domain.
2. Synchronize the system time with the domain controller:
 - a. Open a command prompt.
 - b. Type `Net Time /set /y` and press Enter.
3. Copy the `adtest.msi` file to the server, and double-click to execute.
4. Proceed through the installation, accepting all default options.
5. Repeat these steps on the additional client.

Configuring Active Directory to use ADTest

1. To create the Active Directory structure, run the following command from the command prompt on the server:

```
adtest -r newroot -root 0 -t 20 -sf -o newroot.log
```

2. To add users, we had to make the following changes to existing configuration files:
 - a. Edit the `adtest.ats` file in the install directory and add `//` to comment out the following line:

```
//ATTR unicodePwd:$Password
```

- b. Add the default number of users, 100,000. Type the following command at the command prompt:

```
adtest -r user -root 0 -t 20 -sf -o user.log -user administrator
-password PasswordHere
```

3. Set user passwords. Type the following at the command prompt:

```
adtest -r initpwd -root 0 -t 20 -sf -o user.log -user administrator
-password PasswordHere
```

4. Add Global Groups. Type the following at the command prompt:

```
adtest -r addacc -root 0 -t 20 -sf -o ggroups.log -user
administrator -password PasswordHere
```

5. Add users to the groups. Type the following at the command prompt:

```
adtest -r memacc -set GROUP=0 -root 0 -t 20 -sf -o user2groups.log -
user administrator -password PasswordHere
```

Configuring the database workload

DVD Store Data generation overview

We built the database schema using the scripts in the DS2 distribution package, though we made a few minor modifications. The DS2 stress tool provides options to generate 10MB, 1GB, or 100GB datasets. To get the tool to generate the 5 GB of user data we used in this test, we had to make a few straightforward changes to the source code and to the DVD Store application's scripts. Note: We created our test data on a Linux system to take advantage of the larger RAND MAX.

Editing the ds2_create_orders.c module

The module ds2_create_orders.c defines constants that define the maximum values for the customer ID and the product ID. The constants for the 5GB database size did not exist. We added the constants for this size.

On the command line for the ds2_create_orders.c module, we specified the size. The available options were S (small), M (medium), and L (large). We added the case V for the 5GB database. In the switch statement that sets the values for the variables max_cust_id and max_prod_id, we added cases that assigned them the proper values for the 5GB database size.

We recompiled the ds2_create_orders.c module on Linux, following the instructions in the header comments. We used the following command line:

```
gcc -o ds2_create_orders ds2_create_orders.c -lm
```

Editing the ds2_create_cust.c module

We had to make the same changes to the ds2_create_cust.c module that we made to the ds2_create_orders.c module. On the command line for the ds2_create_cust.c module, we specified the size. The available options were S (small), M (medium), and L (large). We added the case V for the 5GB database. In the switch statement that sets the values for the variables max_cust_id and max_prod_id, we added cases that assigned them the proper values for the 5GB database size.

We recompiled the ds2_create_cust.c module on Linux, following the instructions in the header comments. We used the following command line:

```
gcc -o ds2_create_cust ds2_create_cust.c -lm
```

Generating the data for the 5GB database

We used shell scripts to run all four of the executables that generate the data. The distribution did not include shell scripts for the 5GB size. We wrote shell scripts based on the `ds2_create_cust_large.sh` and `ds2_create_orders_large.sh` scripts. The `ds2_create_prod` and `ds2_create_inv` executables did not ship with associated shell scripts, so we created shell scripts using the instructions in the readme files. We ran the shell scripts in the following order to generate the data for the 5GB database:

1. `ds2_create_orders_5gb.sh`
2. `ds2_create_inv_5gb.sh`
3. `ds2_create_prod_5gb.sh`
4. `ds2_create_cust_5gb.sh`

We waited until the processes finished before we moved onto the next step.

Creating the database

We modified the database creation SQL Server scripts in the DVD Store distribution package to build the database schema, which includes the file structure, tables, indices, stored procedures, triggers, and so on. We built a master copy of the 5GB database version for SQL Server 2005, and then used that master copy to restore our test database to the server or virtual machine between each test run. We stored the backup file on the C: drive of each server for quick access.

We followed these steps to create the database:

1. We created the database and file structure using database creation scripts in the DS2 download. We made size modifications specific to our 5GB database and the appropriate changes to drive letters.
2. We created database tables, stored procedures, and objects.
3. We set the database recovery model to bulk-logged to prevent excess logging.
4. We loaded the data we generated into the database.
5. We created indices, full-text catalogs, primary keys, and foreign keys using the database-creation scripts.
6. We updated statistics on each table according to database-creation scripts, which sample 18 percent of the table data.
7. On the SQL Server 2005 instance, we created a `ds2user` SQL Server login.
8. We set the database recovery model back to full.

We made the following changes in the build scripts:

- Because we varied the size of the datasets, we sized the files in our scripts to reflect the database size and the number of files per filegroup. We allowed for approximately 40 percent free space in our database files to ensure that filegrowth activity did not occur during the testing.

- We followed Microsoft's recommendation of having 0.25 to 1 file per filegroup per core. We used four files per filegroup on the server and virtual machines.
- We did not use the DBCC PINTABLE command for the CATEGORIES and PRODUCTS tables, both because Microsoft recommends against this practice and because the commands do nothing in SQL Server 2005.
- We created a SQL Server login we called ds2user and mapped a database user to this login. We made each such user a member of the db_owner fixed database role.
- Using the DVD Store scripts as a reference, we created the full-text catalog and index on the PRODUCTS table manually in SQL Server Management Studio.

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.

Editing the workload script - ds2xdriver.cs module

To use the 5GB database we created earlier, we had to change the following constants:

- In the routine Controller(), we changed the string sizes. We added the V option for the 5GB database size. DS2 uses the sizes string to interpret the db_size_str option.
- In the class Controller, we changed the arrays MAX_CUSTOMER and MAX_PRODUCT. To each, we added values specifying the bounds for the customer and product IDs. The Controller() routine uses these arrays.
- We added a command-line parameter for the database name:

```
-database_name
```

Editing the workload script - ds2sqlserverfns.cs module

We changed the connection string to increase the number of available connections, to not use the default administrator (sa) account, and to include a parameter for the database name. We raised the available connections limit from the default of 100 to 200 to allow room for experimentation. We created a user account we called ds2User and used that account.

The ds2connect routine in the ds2sqlserverfns.cs module defines sConnectionString. We used the following string; the changes we made appear in bold:

```
string sConnectionString = "User ID=ds2User;Initial
Catalog="+dbname+";Max Pool Size=200;Connection Timeout=120;Data
Source=" + Controller.target;
```

Recompiling the ds2sqlserverdriver.exe executable

We recompiled the ds2xdriver.cs and ds2sqlserverfns.cs module in Windows by following the instructions in the header comments. Because the DS2 instructions were for compiling from the command line, we used the following steps:

1. Open a command prompt.
2. Use the `cd` command to change to the directory containing our sources.
3. Run the batch file `C:\Program Files\Microsoft Visual Studio 9.0\Common7\Tools\vsvars32.bat`. This sets up the environment variables.
4. Execute the following command:

```
csc /out:ds2sqlserverdriver.exe ds2xdriver.cs ds2sqlserverfns.cs  
/d:USE_WIN32_TIMER /d:GEN_PERF_CTRS
```

Installing and configuring the database clients

For the DS2 workload, we used a physical client machine to simulate a number of users putting load on the server. For our client, we created a folder we called `C:\ClientShare` to store workload scripts, and shared this folder for access from our controller machine. We installed the .NET 3.5 framework on each client, as the DS2 test executable requires at least .NET 2.0. We created a performance counter log on the client machine to track the number of orders per minute each database instance returned. We followed this process for the installation:

1. Install Microsoft Windows Server 2003 R2 Enterprise x86 Edition Service Pack 2 on the client.
2. Assign a computer name for the database client.
3. For the licensing mode, use the default setting of five concurrent connections.
4. Enter a password for the administrator log on.
5. Select Eastern Time Zone.
6. Use typical settings for the Network installation.
7. Type `Workgroup` for the workgroup. You may join the domain if you wish.
8. Install Windows Updates, .NET 3.5 framework, and copy the DVD Store client executable into the `c:\clientshare` folder.

Configuring the Exchange workload

Installing the mail test client

Each mail server requires one physical test client system. We used the following process to set up a physical test client system:

1. Install Microsoft Windows Server 2003 R2 Enterprise Edition. Use `mailclient` as the system name.
2. Install Windows Server 2003 Service Pack 2 with the default settings.
3. Open the Network Connections window by right-clicking `My Network Places` and selecting `Properties`.
4. Right-click the network connection, and click `Properties`.
5. Highlight `TCP/IP`, and select `Properties`.
6. Select `Use the following IP address`, and enter the correct information.
7. Click `OK` to close the two `Properties` windows.
8. Right-click `My Computer`, and select `Properties`.
9. Click the `Computer Name` tab.
10. Click `Change`.

11. Select Domain in the member of section, and type the domain name in the text box.
12. In the pop-up window, enter credentials, and click OK.
13. At the Welcome to the domain screen, click OK.
14. When the warning that you must restart dialog appears, click OK.
15. Click OK to close the System Properties window.
16. Click Yes to the pop-up message to restart the computer.
17. Log in to the domain by using administrator and password.
18. Install Microsoft Outlook 2003 with the default settings. (Outlook 2003 does not require any configuration.)
19. Install LoadSim 2003 by double-clicking the loadsim.msi installation file. Accept all defaults.
20. Open LoadSim 2003, and make sure no error messages appear.

Configuring LoadSim 2003

1. Start LoadSim by clicking Start Menu→All Programs→Exchange LoadSim→LoadSim 2003.
2. From the LoadSim menu bar, select Configuration→Topology Properties.
3. Click the Servers tab.
4. In the left pane, click the + next to the name of this server.
5. In the left pane, select the First Storage Group.
6. In the right pane, set the number of users to 300.
7. Select the Security and Credentials tab.
8. Leave the Credentials choice set to Login to users using the account LoadSim is running on.
9. In the password portion of the tab, type the password, and click OK.
10. From the LoadSim menu bar, select Configuration→Test Properties.
11. Click Add, and accept defaults.
12. Click OK.
13. From the LoadSim menu bar, select File→Save As.
14. Save a backup copy of the simulation file as C:\loadsim.sim.
15. From the LoadSim menu bar, select Run→Create Topology, and wait until the Creating Users and Creating DLs tasks complete.
16. From the LoadSim menu bar, select Run→Initialize Test.
17. When a dialog box appears with the question Do you want to initialize Public Folders from this client machine?, click Yes.
18. The initialization is complete when the bottom left corner of the LoadSim window shows Status: Idle.

Backing up the mail database

Once LoadSim has initialized the mail database, you must create a backup copy of this mail database and archive it to a separate location in order to restore cleanly between multiple test runs. Use this process to back up the database:

1. Select Start→All Programs→Microsoft Exchange→System Manager in the mail server.
2. In the left pane, expand Servers until First Storage Group appears highlighted.
3. In the right pane, right-click Mailbox Store, and select Dismount Store from the menu.
4. At the Do you want to continue? pop-up message, click Yes. A red arrow appears in the window pane when you've dismounted the Mailbox Store.

5. Right-click Public Folder Store, and select Dismount Store from the menu.
6. At the Do you want to continue? pop-up message, click Yes. The red arrow appears when you've dismounted the Public Folder Store.
7. Using Windows Explorer, create a new folder <driveletter>:\backup.
8. With Windows Explorer, copy all files from <exchange-data-drive>:\Program Files\Exchsrvr\mdbdata to <driveletter>:\backup. This process may take several minutes.
9. Close Windows Explorer.
10. In Exchange System Manager, right-click Mailbox Store, and select Mount Store from the menu.
11. At The store was successfully mounted pop-up message, click OK.
12. Right-click Public Folder Store, and select Mount Store from the menu.
13. At The store was successfully mounted pop-up message, click OK.
14. Close Exchange System Manager.

Testing procedure

To perform the tests, we used a series of batch files and the PsExec utility, which allowed remote execution of command-line utilities on each of our test client machines. We stored batch files on each client, either under the C:\ClientShare folder, and we used the PsExec utility to coordinate the test procedures amongst the client machine, server machines or virtual machines, and controller. We used simple file sharing to copy files from machine to machine as necessary.

On the three Dell PowerEdge 6800 legacy 32-bit servers, the testing procedure consisted of the following steps:

1. Execute batch files to clean up prior outputs on clients, the servers, and the controller.
2. On the Dell PowerEdge 6800 running SQL Server 2005, drop the test database.
3. On the Dell PowerEdge 6800 running SQL Server 2005, restore the test database.
4. On the Dell PowerEdge 6800 running Exchange 2003, delete the mail database data files.
5. On the Dell PowerEdge 6800 running Exchange 2003, restore the mail database from backup.
6. Reboot the clients and servers, and wait for a ping response from the all machines involved in testing.
7. After all machines reboot, wait 10 additional minutes for any background tasks to complete.
8. Start the DVD Store workload from the database client, LoadSim 2003 workload from the mail client, and ADTest workloads from the two ADTest clients. (See below for specific steps on starting each workload.)
9. Continue the testing for 30 minutes.
10. Stop the DVD Store, LoadSim 2003, and ADTest workloads.
11. Copy all output files to the controller.
12. Repeat twice more and select the median run based on the data output.

After testing the legacy servers, we performed a P2V conversion and migrated our servers to virtual machines running on the Dell PowerEdge R710, running Windows Server 2008 R2 with Hyper-V. After the P2V conversion was complete, the testing procedure on the Dell PowerEdge R710 consisted of the following steps:

1. Execute batch files to clean up prior outputs on clients, the servers, and the controller.
2. On the virtual machine running SQL Server 2005, drop the test database.
3. On the virtual machine running SQL Server 2005, restore the test database.
4. On the virtual machine running Exchange 2003, delete the mail database data files.
5. On the virtual machine running Exchange 2003, restore the mail database from backup.
6. Reboot the clients, shut down the virtual machines, reboot the host server, and wait for a ping response from the all machines involved in testing.
7. After the host reboots, start the virtual machines.
8. After all machines reboot, wait 10 additional minutes for any background tasks to complete.
9. Start the DVD Store workload from the database client, LoadSim 2003 workload from the mail client, and ADTest workloads from the two ADTest clients. (See below for specific steps on starting each workload.)
10. Continue the testing for 30 minutes.
11. Stop the DVD Store, LoadSim 2003, and ADTest workloads.
12. Copy all output files to the controller.
13. Repeat twice more and select the median run based on the data output.

Executing ADTest

On each of the two ADTest clients, launch the ADTest workload, running two threads with the Interactive Login script by executing the following command:

```
"c:\program files\windows resource kits\tools\adtest" -r inter -root 0 -t 2 -o outputlog.log -user administrator -password PasswordHere
```

Executing DVD Store

On the database test client, at a command prompt, run the following command:

```
c:\<path>\ds2sqlserverdriver.exe --target=ServerIPAddress --ramp_rate=10 --run_time=30 --n_threads=30 --db_size_str=V --think_time=0.1 --database_name=DS2
```

(Your parameters may differ based on your workload. In our case, we ran 30 execution threads, each specifying 0.1 seconds of think time.)

Executing LoadSim

1. On the mail test client, launch LoadSim 2003 from Start→All Programs→Exchange LoadSim.
2. On the Menu, select Run→Run Simulation to execute testing.
3. To generate viewable results, run the following from the command prompt in the LoadSim directory:
`lslog.exe answer lspperf.log > test.log` where test.log is the output file. The weighted average in the 95th percentile column is the output we report.

APPENDIX A – SERVER CONFIGURATION INFORMATION

Figure 9 provides detailed configuration information for the test servers.

Servers	Dell PowerEdge 6800 with Intel Xeon Processor MP (Active Directory server)	Dell PowerEdge 6800 with Intel Xeon Processor MP 7020 (database and email servers)	Dell PowerEdge R710 with Intel Xeon Processor X5570 (virtualization server)
General processor setup			
Number of processor packages	2	2	2
Number of cores per processor package	1	2	4
Number of hardware threads per core	2	2	2
CPU			
Vendor	Intel	Intel	Intel
Name	Xeon MP	Xeon MP 7020	Xeon X5570
Stepping	E0	A0	D0
Socket type	PPGA604	PPGA604	FCLGA1366
Core frequency (GHz)	3.16	2.66	2.93
Bus frequency	667 MHz	667 MHz	6.4 GT/s
L2 cache	1 MB	2 MB	4 MB
L3 cache	N/A	N/A	8 MB
Thermal design power (TDP, in watts)	110	165	95
Platform			
Vendor and model number	Dell PowerEdge 6800	Dell PowerEdge 6800	Dell PowerEdge R710
Motherboard model number	0FD006	ORD317	C53661
Motherboard chipset	Intel E8500	Intel E8500	Intel 5520
BIOS name and version	Dell A06 (10/08/2008)	Dell A06 (10/08/2008)	Dell Inc. 1.3.6 (10/30/2009)
BIOS settings	Default	Default	Default
Memory modules			
Vendor and model number	Nanya NT1GT72U4PA0BU-5A	Nanya NT1GT72U4PA0BU-5A	Crucial CT51272BB1339.36SFD1
Type	PC2-3200R ECC DDR2	PC2-3200R ECC DDR2	PC3-10600R DDR3
Speed (MHz)	333	333	1,333
Speed in the system currently running @ (MHz)	667	667	1,066

Servers	Dell PowerEdge 6800 with Intel Xeon Processor MP (Active Directory server)	Dell PowerEdge 6800 with Intel Xeon Processor MP 7020 (database and email servers)	Dell PowerEdge R710 with Intel Xeon Processor X5570 (virtualization server)
Size (GB)	4	4	24
Number of RAM modules	4 x 1 GB	4 x 1 GB	6 x 4 GB
Chip organization	Double-sided	Double-sided	Double-sided
Hard disk			
Vendor and model number	Fujitsu MAT3073NC	Seagate ST373207LC	Seagate ST9146852SS
Number of disks in each system	2	5	8
Size (GB)	73	73	146
Buffer size (MB)	8	8	16
RPM	10,000	10,000	15,000
Type	SCSI	SCSI	SAS 3GB/s
Controller	Dell PERC 4e/DC	Dell PERC 4e/Di	Dell PERC 6/i
Operating system			
Name	Windows Server 2003 R2 Enterprise	Windows Server 2003 R2 Enterprise	Windows Server 2008 R2 Enterprise
Build number	3790	3790	7600
File system	NTFS	NTFS	NTFS
Language	English	English	English
Network card/subsystem			
Vendor and model number	Broadcom BCM5570 NetXtreme Gigabit Ethernet	Broadcom BCM5570 NetXtreme Gigabit Ethernet	Broadcom BCM5709C NetXtreme II Gigabit Ethernet
Type	Integrated	Integrated	Integrated
USB ports			
Number	4	4	4
Type	USB 2.0	USB 2.0	USB 2.0

Figure 9: Detailed configuration information for the test servers.

ABOUT PRINCIPLED TECHNOLOGIES



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

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

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

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

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

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

Disclaimer of Warranties; Limitation of Liability:

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

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