

Executive summary

We compared the database performance and hardware cost of two servers in similar common enterprise configurations running eight virtual machines (VMs). On our tests of Microsoft® SQL Server® 2008 database orders per minute (OPM), the Dell™ PowerEdge™ R710 server delivered multiple advantages over the HP ProLiant DL380 G6 server.



Figure 1: Normalized price-performance results. Higher numbers are better.

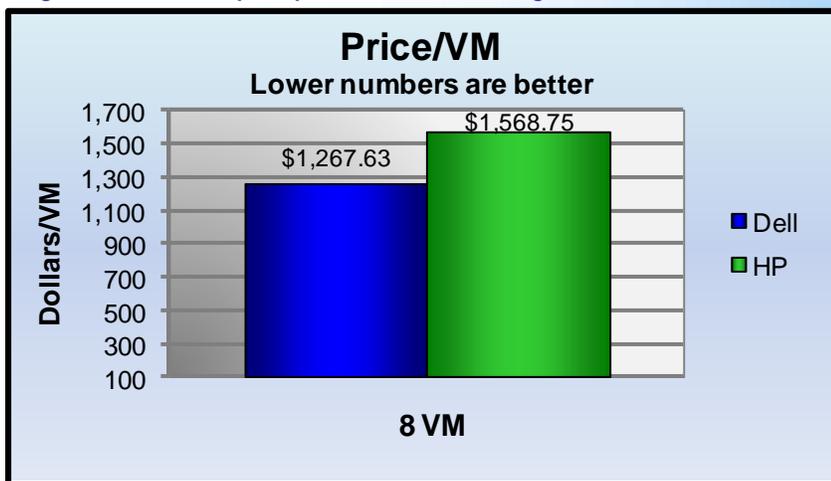


Figure 2: Virtualized database price/VM results. Lower numbers are better.

KEY FINDINGS

The Dell PowerEdge R710 server delivered **20.5% better price-performance** with eight VMs than the HP ProLiant DL380 G6 server.

The Dell PowerEdge R710 server delivered **19.2% better price/VM performance** with eight VMs than the HP ProLiant DL380 G6 server.

What we tested

Our objective was to determine the server's database performance and price advantage while running multiple VMs. We allocated two vCPU to each VM, so we used a total of eight VMs for testing, as both systems under test had a total of 16 logical processors.

We conducted our testing using DVD Store Version 2 (DS2) to generate realistic e-commerce transactional streams where customers log in, browse, and order products from an online DVD store. DS2 is an open-source application with a back-end database component, a front-end Web application layer, and a driver layer that operates as the middle tier and actually executes the workload. We measure DS2 performance in terms of orders OPM, the workload's main reporting metric. The workload calculates OPM as the number of overall orders processed during each test, divided by the elapsed seconds multiplied by 60. 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. The workload also creates new customers. Browsing movies involves select operations, some of which may involve SQL Server 2008 full-text search. The purchase, login, and new customer stored procedures involve execution of database update, insert, and select statements.

Because our goal was to isolate and test database server performance, we did not use the front-end Web application layer. Instead, we ran the driver application on client machines directly via its command-line interface. We used the default DS2 parameters and setup configuration, with the exceptions we note in the DVD Store setup section in the Test methodology section of this report.

Figure 3 shows the system configuration overview for the two similarly configured Dell and HP servers.

Servers	Dell PowerEdge R710	HP ProLiant DL380 G6
Processors	Dual Intel Xeon® Processor E5540, 2.53 GHz	Dual Intel Xeon Processor E5540, 2.53 GHz
Memory	6 x 8GB PC3-8500R DDR3	6 x 8GB PC3-8500R DDR3
Hard disks	2 x 73GB, 15KRPM, SAS	2 x 73GB, 15KRPM, SAS
Power supplies	2 Dell A570P-00, 570W	2 HP HSTNS-PL14, 460W
Disk controller	Dell PERC 6/i Integrated RAID Controller, 256MB cache and BBWC	HP 410i Integrated Smart Array Controller, 256MB cache and BBWC
Operating system	Windows Server® 2008 R2 Enterprise	Windows Server 2008 R2 Enterprise
Database	Microsoft SQL Server 2008 SP1	Microsoft SQL Server 2008 SP1

Figure 3: System configuration overview for the two test servers. See Appendix A for more details.

For client machines, we used Dell PowerEdge M600 blade servers running Microsoft Windows Server® 2003, Enterprise Edition. Each of these clients ran a single instance of DS2 that spawned 32 threads. To simulate a heavily loaded environment, the load-generating client systems ran with no think time, issuing requests as quickly as the server could handle them. Our goal was to maximize processor utilization for the given server configurations using a demanding workload.

The DS2 driver application creates an orders-per-minute performance counter on each client machine. The DVD Store client application visually outputs OPM at 10-second intervals, but we chose to collect this OPM metric via a performance monitor counter on each client machine at 1-second intervals.

We ran this workload on both servers for 30 minutes.

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

What we found

Figure 4 illustrates the performance in OPM, by VM and total, for the Dell PowerEdge R710 server and the HP ProLiant DL380 G6 server. The results are the median of three runs. Higher OPM numbers are better.

Median performance results		
	Dell PowerEdge R710	HP ProLiant DL380 G6
1 VM OPM	13,916	14,116
2 VM OPM	14,622	13,560
3 VM OPM	13,910	14,200
4 VM OPM	14,264	14,245
5 VM OPM	14,407	14,430
6 VM OPM	14,364	14,290
7 VM OPM	14,496	14,187
8 VM OPM	14,710	13,835
Total OPM	114,689	112,863

Figure 4: Median performance results for the Dell PowerEdge R710 server and the HP ProLiant DL380 G6 server. Higher OPM numbers are better.

Figure 5 shows the price-performance gain of the Dell PowerEdge R710 server over the HP ProLiant DL380 G6 server on the median runs. The Dell PowerEdge R710 delivered 20.5 percent better price-performance than the HP ProLiant DL380 G6. The Dell PowerEdge R710 server's retail price is \$10,141 and the HP ProLiant DL380 G6 server's retail price is \$12,550. Lower dollars/OPM numbers are better.

To calculate the price-performance, we used the following formula:

$$(\text{Price of server}) / (\text{Total OPMs supported}) = \text{price-performance}$$

Price-performance results	
	Dollars/OPM
Dell PowerEdge R710	\$0.088
HP ProLiant DL380 G6	\$0.111

Figure 5: Price-performance results for the Dell PowerEdge R710 server and the HP ProLiant DL380 G6 server while running the DVD Store workload. Lower dollars/OPM numbers are better.

Figure 6 shows the price/VM results of the Dell PowerEdge R710 server and the HP ProLiant DL380 G6 server. The Dell PowerEdge R710 delivered 19.2 percent better price/VM than the HP ProLiant DL380 G6.

To calculate the price/VM, we used the following formula:

$$(\text{Price of server}) / (\text{Total VMs supported}) = \text{price/VM}$$

Price/VM results	
	Dollars/VM
Dell PowerEdge R710	\$1,267.63
HP ProLiant DL380 G6	\$1,568.75

Figure 6: Price/VM results for the Dell PowerEdge R710 server and the HP ProLiant DL380 G6 server while running the DVD Store workload. Lower price/VM numbers are better.

How we tested

Adjusting BIOS settings

We used the default BIOS settings on both the Dell PowerEdge R710 server and HP ProLiant DL380 G6 server, except that we disabled hardware prefetch and node interleaving.

Setting up and configuring the servers and storage

We installed two internal drives in the Dell PowerEdge R710 and HP ProLiant DL380 G6 servers. We configured these drives as a RAID 1 array for the Microsoft Server 2008 R2 installation on both systems. We connected four Dell EqualLogic™ PS5000XV arrays to the Dell PowerEdge R710 to store the VMs. We attached the HP ProLiant DL380 G6 server to a HP StorageWorks EVA4400 with five shelves to store the VMs.

Configuring the Dell Storage

Each of the four Dell EqualLogic arrays has 16 drives, for a total of 64 drives. Each of the five HP StorageWorks EVA4400 storage trays has 12 drives, for a total of 60 drives. To match the drive count of the two storage configurations, we configured two Dell EqualLogic arrays in RAID 10 no-spares mode (16 active drives each) and two Dell EqualLogic arrays in regular RAID 10 mode (14 active drives each), for a total of 60 active drives.

We connected the Dell PowerEdge R710 server to the Dell EqualLogic storage via three 1 Gb/s Ethernet cables and three 1 Gb/s iSCSI ports on each of the Dell EqualLogic PS5000XV arrays. We connected both to a Dell PowerConnect™ 6248 switch. We configured the Dell EqualLogic storage into two storage pools. The VM operating system and log storage pool contained one Dell EqualLogic PS5000XV array. The database storage pool contained three Dell EqualLogic PS5000XV arrays. We created one 900GB volume in the VM operating system and log storage pool, and one 1.99TB volume in the database storage pool. We used the following steps to configure the storage:

1. Using the command line, set up a storage group with the following RAID policies:

- PS5000XV-1: RAID-10
 - PS5000XV-2: RAID-10 (no spares)
 - PS5000XV-3: RAID-10
 - PS5000XV-4: RAID-10 (no spares)
2. Create a storage pool you name `OS-Log` with the following member:
 - PS5000XV-1
 3. Create a storage pool you name `database` with the following members:
 - PS5000XV-2
 - PS5000XV-3
 - PS5000XV-4
 4. Enable all network connections on each Dell EqualLogic PS5000XV using the following IP address scheme:
 - IP Address: `192.168.1.##` (## being any number between 10 and 60)
 - Subnet Mask: `255.255.255.0`
 5. Create one 900GB volume in the OS-Log storage pool with no snapshot reserve, and name it `OS-Log`
 6. Create one 1.99TB volume in the database storage pool with no snapshot reserve, and name it `database`
 7. Enable shared access to the iSCSI target from multiple initiators on every volume.
 8. Create an access control record for every volume without specifying any limitations.

Configuring the HP storage

We connected the HP ProLiant DL380 G6 server to the HP StorageWorks storage via two 4 Gb/s Fibre Channel cables into a Brocade 200E 4Gb Fibre Channel switch. We connected the HP StorageWorks storage into the Brocade 200E Fibre Channel switch using four 4 Gb/s Fibre Channel Cables. We configured the HP StorageWorks into two disk groups using RAID 10. Our database disk group contained 46 disks. Our VM operating system and log disk group contained 14 disks. We created one 955GB volume in the operating system and log disk group, and one 2,047 GB volume in the database disk group. We used the following steps to configure the storage:

1. Create a disk group with the following parameters:
 - Name: Database
 - Disks: 1-46
 - Disk drive failure protection: None
 - Alarm Level: 100%
2. Create a disk group with the following parameters:
 - Name: OS-Log
 - Disks: 47-60
 - Disk drive failure protection: None
 - Alarm Level: 100%
3. Create a new vDisk with the following parameters:
 - Name: Database1
 - Size: 2,047 GB
 - Disk Group: Database
 - Redundancy: Vraid 1
4. Create a new vDisk with the following parameters
 - Name: OS-Log
 - Size: 955 GB
 - Disk Group: OS-Log
 - Redundancy: Vraid 1
5. Create a Host, and add the world wide name for both Fibre Channel adapters on the server for each vDisk.

Setting up and configuring the OS and VMs

We performed the following steps on both the Dell PowerEdge R710 server and HP ProLiant DL380 G6 server to install the OS and VMs for testing.

Installing Windows Server 2008 R2

1. Insert the installation DVD for Windows Server 2008 R2 into the DVD drive.

2. Choose the language, time and currency, and keyboard input. Click Next.
3. Click Install Now.
4. Choose Windows Server Enterprise (Full Installation). Click Next.
5. Accept the license terms, and click Next.
6. Click Custom.
7. Click the Disk, and click Drive options (advanced).
8. Click New, Apply, Format, and click Next.
9. After the installation completes, click OK to set the Administrator password.
10. Enter the administrator password twice, and click OK.
11. Click Start, type `change power-saving settings` and press Enter.
12. Click Change plan settings.
13. Change the Turn off the display drop-down menu to Never.
14. Click Save changes, and close the Power Options, Screen Saver Settings, and Personalization windows.

Installing system updates in Windows Server 2008 R2

We installed the following updates on the system using the Microsoft Windows Update feature:

- Security Update for Windows Server 2008 R2 x64 Edition (KB974571)
- Update for Windows Server 2008 R2 x64 Edition (KB974431)
- Security Update for Internet Explorer 8 for Windows Server 2008 R2 x64 Edition (KB974455)
- Security Update for ActiveX killbits for Windows Server 2008 R2 x64 Edition (KB973525)
- Security Update for Windows Server 2008 R2 x64 Edition (KB975467)

Installing the Hyper-V role in Windows Server 2008 R2

1. Click Start→Administrative Tools→Server Manager→Roles→Add Roles.
2. Select the Hyper-V role by checking the box beside it, and click Next.
3. Check the box beside the Ethernet Card you wish to use for client/server traffic, and click Next.
4. Click Install to begin the Hyper-V installation.
5. When the installation finishes, click Close.
6. When the installer prompts you to restart now, click Yes.
7. Once the system finishes rebooting (this might happen twice), log in. The Resuming Configuration Wizard should start automatically.
8. Once the Resuming Configuration Wizard completes, the installation results should appear with a message that the installation succeeded.
9. Click Close.
10. Restart the server.

Creating the Hyper-V virtual machine

1. Click Start→Administrative Tools→Hyper-V Manager.
2. From the Action menu, select New Virtual Hard Disk.
3. Choose Fixed Size, assign 45 GB for size, and place the VHD in the appropriate VM folder.
4. From the Action menu, select New Virtual Machine.
5. On the Before You Begin page, click Next.
6. Enter a name for the virtual machine, and browse to the location where you want to store the VM files. (The New Virtual Machine Wizard will automatically create a subfolder with the name of the VM.)
7. On the Assign Memory page, enter 5,120 MB of RAM.
8. On the Configure Networking page, choose the network assigned to client/server traffic, and click Next.
9. Select Attach a virtual hard disk later, and click Next.
10. On the installation Options page, accept the default of Install an operating system later, and click Next.
11. On the Completing the New Virtual Machine Wizard page, click Finish.

Modifying the virtual machine

1. Right-click the virtual machine, and choose Settings.
2. Click Processors, and choose 2 virtual processors.
3. Click the virtual IDE controller 0.
4. Click Add, and click Hard Drive.
5. Browse to the VHD you created in Step 3, and choose that VHD for the guest OS VHD.
6. Click Apply.
7. Click the SCSI Controller, and click the Physical hard disk radio button to configure your SQL Server data and SQL Server log pass-through disks. You may wish to do this step after cloning VMs.

8. Choose the appropriate LUN.
9. Repeat steps 7 and 8 for the remaining SQL Server data and SQL Server log LUNs you assigned to this VM.
10. After you have cloned out your VMs, repeat steps 1 through 9 for all remaining VMs.

Installing the guest operating system

1. Insert the installation DVD for Windows Server 2008 R2 into the DVD drive.
2. In Hyper-V Manager, right-click the virtual machine, and choose Connect.
3. On the media menu, choose DVD DriveCapture D:
4. On the Action menu, choose Start.
5. Choose the language, time and currency, and keyboard input. Click Next.
6. Click Install Now.
7. Choose Windows Server Enterprise (Full Installation), and click Next.
8. Accept the license terms, and click Next.
9. Click Custom.
10. Click the Disk, and click Drive options (advanced).
11. Click New, Apply, Format, and click Next.
12. Let the installation process continue. The VM will reboot several times.
13. After the installation completes, click OK to set the Administrator password.
14. Enter the administrator password twice, and click OK.
15. When the Windows installation completes, select Action→Insert Integration Services Setup Disk from the menu. Installation should begin automatically.
16. At the prompt to restart, click Yes.
17. Once the system has restarted, log in, and verify that all configured devices (mouse, keyboard, video, network, etc.) are working.
18. Once you are sure all your devices are working, from the menu, select Media→DVD Drive→Eject vmguest.iso.

Installing Windows updates

1. Click Start→ Windows Update.
2. Click View Advanced Options.
3. Click Never Check for Updates.
4. Click OK.
5. Click Check for Updates.
6. Click Install Now when the application prompts you to install an update for Windows Update.
7. Click View available updates to ensure you install the necessary updates.
8. Click Install.
9. Agree to the license terms, and click Finish.
10. Reboot the VM when the application requests you do so.
11. After reboot, check Windows Update again for further updates. Click Start→Windows Update.
12. Click Check for updates.
13. If Windows lists no important updates, close the updates window.

Installing SQL Server 2008 on the VM

1. Insert the installation DVD for SQL Server 2008 into the DVD drive.
2. In Hyper-V Manager, right-click the virtual machine, and choose Connect.
3. On the media menu, choose DVD DriveCapture D:
4. On the Action menu, choose Start.
5. Click Run SETUP.EXE on the AutoPlay menu.
6. If the application prompts you to install the .NET Framework, click OK.
7. Select the Accept radio button, and click Install.
8. When the installation finishes, click Exit.
9. If any warnings appear about compatibility, accept them, as installing SP1 in the next section should solve them.
10. If the application prompts you, click OK to install a hotfix for Windows (KB942288).
11. Click Restart now to restart the server upon completion.
12. After rebooting, log into the server.
13. Click Start, and click Computer.
14. Double-click the CD/DVD drive.

15. Click Installation.
16. Click New SQL Server stand-alone installation.
17. On the Setup support rules screen, click OK.
18. Specify the Enterprise Evaluation free edition. Click Next.
19. Click the checkbox to accept the license terms. Click Next.
20. Click Install to install the setup support files.
21. You may see a Windows Firewall warning. For now, ignore this, and click Next.
22. On the Feature Selection screen, select Database Engine Services, Full-Text Search, Client Tools Connectivity, Client Tools Backward Compatibility, Management Tools –Basic, and Management Tools – Complete. Click Next.
23. On the Instance configuration screen, leave the default selection of default instance, and click Next.
24. On the Disk space requirements screen, click Next.
25. On the Server configuration screen, choose NT AUTHORITY\SYSTEM for SQL Server Agent, and choose NT AUTHORITY\SYSTEM for SQL Server Database Engine. Click Next.
26. On the Database Engine Configuration screen, select Mixed Mode.
27. Enter a password for the system administrator (sa) account.
28. Click Add Current user. This may take several seconds.
29. Click Next.
30. On the Error and usage-reporting screen, click Next.
31. On the Installation rules screen, click Next.
32. On the Ready to Install screen, click Next.
33. After installation completes, click Start, All Programs, Microsoft SQL Server 2008, Configuration Tools, SQL Server Configuration Manager.
34. Expand the Network Configuration node, and select the relevant instance.
35. In the right pane, right-click TCP/IP, and choose Enable.
36. In the left pane, click SQL Server Services.
37. In the right pane, right-click the SQL Server service, and select Restart.
38. Create a SQL Server login for the ds2user (see the Setting up DVD Store section below for the specific script to use).
39. Place the DVD Store backup file you prepared in the default location for SQL Server: C:\Program Files\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\Backup.

Installing SQL Server 2008 SP1

Service Pack 1 is available for download from

<http://www.microsoft.com/downloads/details.aspx?FamilyID=66ab3dbb-bf3e-4f46-9559-ccc6a4f9dc19&displaylang=en>.

1. Run SQLServer2008SP1-KB968369-x64-ENU.exe.
2. On the Welcome screen, click Next
3. On the License terms screen, accept the license terms, and click Next.
4. On the Features screen, select available features, and click Next.
5. On the Check files in use screen, click Next.
6. On the Update screen, click Next.
7. When the update completes, click Next, and click Close.

Setting up 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 10 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 10GB 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 T for the 10GB 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 10GB 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 T for the 10GB 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 10GB 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 10GB 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 10GB 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 10GB database:

1. ds2_create_orders_10gb.sh
2. ds2_create_inv_10gb.sh
3. ds2_create_prod_10gb.sh
4. ds2_create_cust_10gb.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 10GB database version for SQL Server 2005 and for SQL Server 2008, and then used those master copies to restore our test database to the HP ProLiant DL380 G6 server and Dell PowerEdge R710 server 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 10GB 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. 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.
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 each SQL Server instance, we created a ds2user SQL Server login using the following Transact SQL (TSQL) script:

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

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 logical processor. We used two files per filegroup in each VM on the Dell PowerEdge R710 and HP ProLiant DL380 G6 server.
- 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 and SQL Server 2008.
- In the case of SQL Server 2008, we added the FORCESEEK hint in the BROWSE_BY_ACTOR stored procedure because in initial testing we found that SQL Server 2008 was choosing an index scan instead of the preferred index seek, adding unnecessary overhead. This problem was not evident 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 10GB database we created earlier, we had to change the following constants:

- In the routine `Controller()`, we changed the string sizes. We added the T option for the 10GB 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 for us.
4. Execute the following command:

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

Conducting the test

To perform the test, we used a series of batch files. We stored batch files either under the C:\ServerShare folder on each server or under the C:\ClientShare folder on the clients, and we used the PsExec utility to coordinate the test procedures amongst the client machines, server machines, and controller. We used simple file sharing to copy files from machine to machine, as needed.

The testing procedure consisted of the following steps:

1. Execute batch files to clean up prior outputs on client machines, the server machine, and the controller.
2. Drop the database in each VM.
3. Restore the database in each VM.
4. Pause briefly to wait for background tasks to complete before the scripts reboot the server.
5. The scripts reboot the client machines, shut down all VMs, reboot the server machine, and wait for a ping response from all test machines.
6. Start all VMs.
7. Wait 10 additional minutes for any background tasks to complete.
8. Start the counter logs on client machines, and start the data collector set on the server under test.
9. Start the workload connections.
10. Start the workload ramp-up period.
11. Start the workload on each VM, and run it for 30 minutes.
12. Stop the workload.
13. Copy all output files to the controller.

We obtained performance test results by running the workload three times under each specific load generation configuration, and by taking the median result as representative of the sample.

Appendix A – Test server information

Figure 7 presents detailed information for the test servers.

Servers	Dell PowerEdge R710	HP ProLiant DL380 G6
General dimension information		
Height (inches)	3.50	3.39
Width (inches)	17.50	17.53
Depth (inches)	27.00	25.81
U size in server rack (U)	2	2
Power supplies		
Total number	2	2
Brand and model	Dell A570P-00	HP HSTNS-PL14
Wattage (W)	570	460
Cooling fans		
Total number	5	6
Dimensions (h x w)	2.5" x 2.5"	2 3/8" x 2.5"
Voltage (V)	12	12
Amps (A)	1.60	2.45
General processor setup		
Number of processor packages	2	2
Number of cores per processor package	4	4
Number of hardware threads per core	2	2
System power management policy	Balanced	Balanced
CPU		
Vendor	Intel	Intel
Name	Xeon E5540	Xeon E5540
Stepping	D0	D0
Socket type	LGA1366	LGA1366
Core frequency (GHz)	2.53	2.53
L1 cache	4 x 32 KB + 32 KB	4 x 32 KB + 32 KB
L2 cache	4 x 256 KB	4 x 256 KB
L3 cache (MB)	8	8
Platform		
Vendor and model number	Dell PowerEdge R710	HP ProLiant DL380 G6
Motherboard model number	0N047H	PADAB0G9VXC1CQ
Motherboard revision number	13	13
BIOS name and version	Dell 1.2.6 (07/17/2009)	HP BIOS P62 (10/01/2009)
BIOS settings	Hardware Prefetch disabled, Node Interleaving disabled	Hardware Prefetch disabled, Node Interleaving disabled
Memory modules		
Total RAM in system (GB)	48	48
Vendor and model number	Samsung M393B1K70BH1	Samsung M393B1K70BH1
Type	PC3-8500R DDR3	PC3-8500R DDR3
Speed (MHz)	1,066	1,066
Speed in the system currently running @ (MHz)	1,066	1,066

Servers	Dell PowerEdge R710	HP ProLiant DL380 G6
Timing/latency (tCL-tRCD-iRP-tRASmin)	7-7-7-20	7-7-7-20
Size (GB)	48	48
Number of RAM modules	6 x 8 GB	6 x 8 GB
Chip organization	Double-sided	Double-sided
Hard disk		
Vendor and model number	Seagate ST973451SS	Fujitsu DH072BAAKN
Number of disks in system	2	2
Size (GB)	73	73
Buffer size (MB)	16	16
RPM	15,000	15,000
Type	SAS	SAS
Controller	Dell PERC 6/i Integrated RAID Controller, 256MB cache and BBWC	HP 410i Integrated Smart Array Controller, 256MB cache and BBWC
Operating system		
Name	Windows Server 2008 R2 Enterprise	Windows Server 2008 R2 Enterprise
Build number	7600	7600
File system	NTFS	NTFS
Language	English	English
Network card/subsystem		
Vendor and model number	Broadcom NetXtreme II gigE 5709C Quad-Port Ethernet	Broadcom NetXtreme II gigE 5709C Quad-Port Ethernet
Type	Integrated	Integrated
Fibre card		
Vendor and model number	N/A	HP StorageWorks FC2242 Dual Channel 4Gb PCI-E HBA
Type	N/A	PCI Express
Optical drive		
Vendor and model number	TEAC DV28SV DVD-ROM	LG GDR-D20N DVD-ROM
USB ports		
Number	4	4
Type	2.0	2.0

Figure 7: Detailed system configuration information for the two test servers.

Appendix B– Test storage information

Figure 8 presents detailed information for the test storage arrays.

	Dell EqualLogic iSCSI SAN	HP StorageWorks Fibre Channel SAN
Arrays	Three Dell EqualLogic PS5000XV arrays and one Dell EqualLogic PS6000 Array	One HP StorageWorks EVA 4400 Enterprise Virtual Array with five full disk shelves
Disks	60 active 15,000RPM SAS disks total for the four shelves	60 active 15,000RPM FC disks total for the five shelves
Active storage cache	4 GB (1 GB per Dell EqualLogic PS5000XV array)	4 GB
Number of active storage controllers	4 (1 per Dell EqualLogic PS5000XV array)	2
Number of passive storage controllers	4 (1 per Dell EqualLogic PS5000XV array)	0
Number of active storage ports	12 (3 per Dell EqualLogic PS5000XV array)	4
Switch type/ model/firmware revision	Dell PowerConnect 6248 v2.2.0.3	Brocade 200E Fabric OS v6.0.0b
Disk vendor and model number	48 x Seagate ST3146855SS 16x Maxtor ATLAS15K2_147SAS	36x HP BDF146DA47A 24x HP BF146DA47A
Disk size (GB)	146	146
Disk buffer size (MB)	16	16
Disk RPM	15,000	15,000
Disk type	SAS	Fibre Channel

Figure 8: Detailed configuration information for the storage arrays.

About Principled Technologies

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 you assess how it will fare against its competition, its performance, whether it's ready to go to market, 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, Inc.
1007 Slater Rd., Suite 250
Durham, NC 27703
www.principledtechnologies.com
info@principledtechnologies.com

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.