Principled Technologies®

TEST REPORT
APRIL 2009

OLTP performance comparison: Intel Xeon Processor E5506-based Dell PowerEdge R710 server vs. Intel Xeon Processor E5520-based Dell PowerEdge R710 server

Executive summary

Dell Inc. (Dell) commissioned Principled Technologies (PT) to compare the Microsoft SQL Server 2008 active online transaction processing (OLTP) performance of two Intel Xeon processors, each running on the same Dell PowerEdge R710 server:

- Intel Xeon Processor E5506 (2.13GHz Quad-Core Server Processor)
- Intel Xeon Processor E5520 (2.26GHz Quad-Core Server Processor)

KEY FINDING

The Intel Xeon Processor E5520-based Dell PowerEdge R710 server achieved 75.9 percent higher SQL Server 2008 performance than did the Intel Xeon Processor E5506-based Dell PowerEdge R710 server. (See Figure 1.)

The Intel Xeon Processor E5506 has eight cores, for a total of eight logical processors. The Intel Xeon Processor E5520 has eight cores and hyper-threading capabilities, for a total of 16 logical processors.

In our testing, the Dell PowerEdge R710 server contained two of the processors under test and had attached storage consisting of three Dell EqualLogic arrays. We connected the Dell PowerEdge R710 server to the Dell EqualLogic storage arrays via iSCSI. We installed two instances of SQL Server 2008 on the Dell PowerEdge R710 solution and ran one 10GB DS2 database on each instance. Thus, we measured the performance of the overall solution, both server and storage, especially as it relates to an active online transaction processing environment.

We selected the DVD Store Version 2 (DS2) test tool. DS2 is an open-source simulation of an online e-commerce DVD store, where customers log in, browse, and order products. The main DS2 metric is orders per minute (OPM). We report OPM results from a 300-second period of steady activity and heavy load during the DS2 test run. The Workload section of this report explains DS2 in more detail.

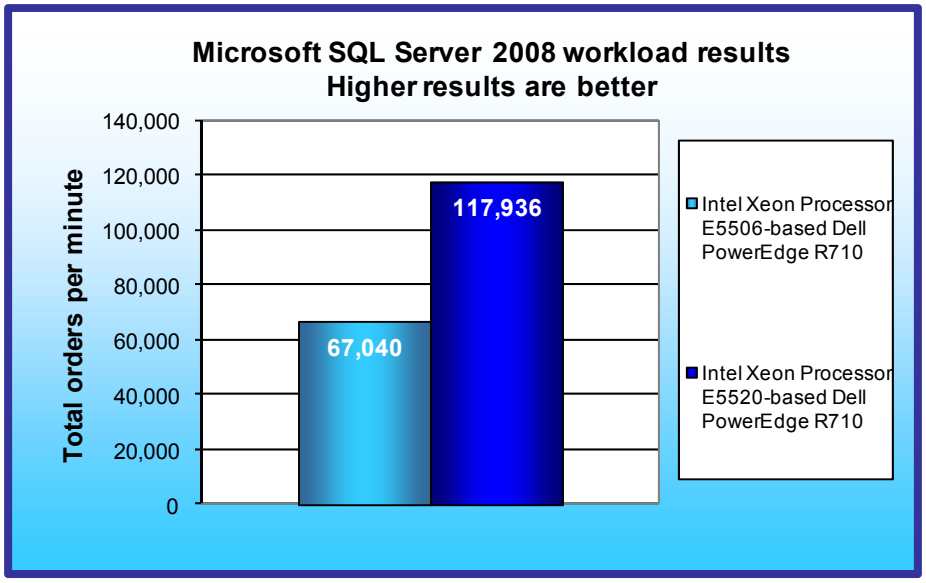

Figure 1: DVD Store results in orders per minute for the Intel Xeon Processor E5506-based Dell PowerEdge R710 and the Intel Xeon Processor E5520-based Dell PowerEdge R710 server. Higher numbers are better.

As Figure 1 shows, the Intel Xeon Processor E5520-based Dell PowerEdge R710 server delivered 75.9 percent more OPM than did the Intel Xeon Processor E5506-based Dell PowerEdge R710 server: 117,936 OPM vs. 67,040 OPM, respectively.

Workload

DVD Store Version 2 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.

Because our goal was to isolate and test database server and

storage, we did not use the Web application layer. Instead, we ran the driver application on client machines directly via its command-line interface.

DS2 models an online DVD store. 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 use SQL Server 2008 full-text search and some of which do not. The purchase, login, and new customer stored procedures involve update and insert statements, as well as select statements. The workload's main reporting metric is orders per minute.

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

For client machines, we used two Dell PowerEdge 2950 servers running Windows Server 2003 Enterprise Edition x86. Each of these two client machines ran a single instance of DS2, which spawned 32 threads. This simulated a heavily loaded environment; the load-generating clients ran with no think time, blasting requests as quickly as the servers could handle them. 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.

The DS2 driver application creates an orders-per-minute performance counter on each client machine. While the DVD Store client application visually outputs OPM at 10-second intervals, 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 the Dell PowerEdge R710 solution for 30 minutes.

Test results

Figure 2 shows the number of OPM each solution achieved during our measurement period, minutes 24 through 29 of the test. We calculated the scores by averaging the OPM for each of the two SQL instances during the 5-minute period and then adding those averages.

Server	Run 1	Run 2	Run 3	Median
Intel Xeon Processor E5506-based Dell PowerEdge R710 server with Dell EqualLogic storage arrays	67,040	67,012	67,190	67,040
Intel Xeon Processor E5520-based Dell PowerEdge R710 server with Dell EqualLogic storage arrays	118,224	117,733	117,936	117,936

Figure 2: OPM scores for the two solutions during minutes 24 through 29 of the DVD Store test. Higher numbers are better.

Test methodology

Setting up and configuring the storage

Dell EqualLogic storage configuration

Each of the three Dell EqualLogic arrays has 16 drives, for a total of 48 drives. We configured each array in RAID 10 mode.

We connected the Dell PowerEdge R710 server to the Dell PowerConnect 6248 switch via three 1Gb/s Ethernet cables. We also connected the Dell EqualLogic storage to the Dell PowerConnect 6248 switch. We configured the Dell EqualLogic storage into one storage pool. We spread the database storage pool (Storage Pool 1) across the three Dell EqualLogic PS5000XV Arrays. We created four 80GB volumes in the database storage pool. We used the Microsoft iSCSI Initiator with the EqualLogic Host Integration Toolkit (HIT) on the Dell PowerEdge R710 server to connect to each 80GB volume. We placed the SQL Server transaction logs on four internal drives configured in a RAID 0 array.

Setting up the internal storage

1. Enter the RAID controller BIOS by pressing Ctrl+R.

2. Create a new RAID 1 volume for the OS by highlighting Controller 0, pressing F2, and selecting Create New VD (Virtual Disk).
3. Choose physical disks, and set configuration options for the virtual disk.
4. Initialize the volume by selecting the virtual disk and pressing F2→Initialization→Start Init.
5. Repeat steps 2 through 3 for the second volume, and choose a 4 disk RAID 0.

Setting up the external storage

1. Using the command line, setup a storage group with the following RAID policies:
    a. PS5000XV-1: RAID-10
    b. PS5000XV-2: RAID-10
    c. PS5000XV-3: RAID-10
2. Create a storage pool that you name `database` with the following members:
    a. PS5000XV-1
    b. PS5000XV-2
    c. PS5000XV-3
3. Enable all network connections on each PS5000XV using the following IP address scheme:
    a. IP Address: `192.168.1.##` (## being any number between 10 and 60)
    b. Subnet Mask: `255.255.248.0`
4. Create four 80GB volumes in the database storage pool with no snapshot reserve, and name them `dvd-store#` (# being 1 through 4).
5. Enable shared access to the iSCSI target from multiple initiators on every volume.
6. Create an access control record for every volume without specifying any limitations.
7. Create a read-only SNMP community name to use for group monitoring.

Adjusting BIOS settings

We used the default BIOS settings.

Setting up the server

We installed a fresh copy of Windows Server 2008, Enterprise x64 Edition on the Dell PowerEdge R710.

Installing Windows Server 2008 Enterprise Edition x64

1. Boot the server, and insert the Windows Server 2008 x64 installation DVD in the DVD-ROM drive.
2. At the Language Selection Screen, click Next.
3. Click Install Now.
4. Select Windows Server 2008 Enterprise (Full Installation) x64, and click Next.
5. Click the I accept the license terms check box, and click Next.
6. Click Custom.
7. Click Drive options (advanced).
8. Click New.
9. Click Apply.
10. Click Next.
11. At the User's password must be changed before logging on warning screen, click OK.
12. Type `Password1` as the new password in both fields, and click the arrow to continue.
13. At the Your password has been changed screen, click OK.

Setting up network configuration on the server

1. Click Start→Control Panel→Network and Internet→Network Connections, and double-click the Local Area Connection assigned to network traffic.
2. Select Internet Protocol Version 4 (TCP/IPv4), and click Properties.
3. In the Internet Protocol Version 4 (TCP/IPv4) Properties screen, select the Use the following IP address radio button.
4. Enter a valid static IP address, subnet mask, and default gateway.
5. Select the Use the following DNS server addresses radio button, and enter at least one valid DNS server address.

6. Click OK, and click Close to exit.
7. Repeat steps 1 through 6 for the remaining three Network Connections you will use for iSCSI, modifying the subnet mask to match the storage subnet mask.

Installing system updates in Windows Server 2008 x64

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

- Hotfix for Microsoft Windows (KB942288)
- Update for Microsoft Windows (KB967190)
- Security Update for Microsoft Windows (KB958624)
- Security Update for Microsoft Windows (KB958644)
- Security Update for Microsoft Windows (KB958687)
- Security Update for Microsoft Windows (KB958690)
- Security Update for Microsoft Windows (KB960225)
- Security Update for Microsoft Windows (KB960715)
- Security Update for Microsoft Windows (KB961260)
- Update for Microsoft Windows (KB959130)
- Hotfix for Microsoft Windows (KB949189)
- Hotfix for Microsoft Windows (KB952287)
- Security Update for Microsoft Windows (KB938464)
- Security Update for Microsoft Windows (KB950760)
- Security Update for Microsoft Windows (KB950762)
- Security Update for Microsoft Windows (KB950974)
- Security Update for Microsoft Windows (KB951066)
- Security Update for Microsoft Windows (KB951698)
- Security Update for Microsoft Windows (KB953733)
- Security Update for Microsoft Windows (KB954459)
- Security Update for Microsoft Windows (KB955069)
- Security Update for Microsoft Windows (KB956802)
- Security Update for Microsoft Windows (KB956841)
- Security Update for Microsoft Windows (KB957097)
- Security Update for Microsoft Windows (KB958623)
- Update for Microsoft Windows (KB940518)
- Update for Microsoft Windows (KB948609)
- Update for Microsoft Windows (KB948610)
- Update for Microsoft Windows (KB950050)
- Update for Microsoft Windows (KB951978)
- Update for Microsoft Windows (KB955020)
- Update for Microsoft Windows (KB955302)
- Update for Microsoft Windows (KB955839)
- Update for Microsoft Windows (KB957200)
- Update for Microsoft Windows (KB957321)
- Update for Microsoft Windows (KB957388)
- Update for Microsoft Windows (KB958481)
- Update for Microsoft Windows (KB958483)
- Hotfix for Microsoft .NET Framework 3.5 SP1 (KB953595)
- Hotfix for Microsoft .NET Framework 3.5 SP1 (KB958484)
- Hotfix for Microsoft Visual Studio 2007 Tools for Applications – ENU (KB946040)
- Hotfix for Microsoft Visual Studio 2007 Tools for Applications – ENU (KB946308)
- Hotfix for Microsoft Visual Studio 2007 Tools for Applications – ENU (KB946344)
- Hotfix for Microsoft Visual Studio 2007 Tools for Applications – ENU (KB947540)
- Hotfix for Microsoft Visual Studio 2007 Tools for Applications – ENU (KB947789)

Connecting the storage to the host

We attached the four database volumes on the Dell EqualLogic storage to the Dell PowerEdge R710 server by connecting all three NICs available on each PS5000XV array to a Dell PowerConnect 6248 switch. We used three physical NICs in the server dedicated to iSCSI traffic to connect the Microsoft iSCSI Initiator within the server to the four 80GB database volumes. We used the remaining NIC in the server for normal network communication, and connected this NIC to the network via a Dell PowerConnect 5448 switch. iSCSI traffic used one switch and subnet, while normal network communication used a separate switch and subnet.

Adding an additional drive to the server via the iSCSI initiator

1. Click Start, Control Panel, and double-click iSCSI Initiator.
2. Click the Discovery tab.
3. Click Add Portal.
4. Add the IP address of the Dell EqualLogic storage group.
5. Click OK.
6. Click the Targets tab.
7. Find the relevant SQL Data volume for this server, and select it. Click Log On…
8. Check the checkbox to allow automatic logon on reboot and to enable multipath.
9. Click OK.
10. Repeat steps 7 through 9 for the remaining volumes.

Installing SQL Server 2008 on the server

1. Insert the DVD into the physical DVD drive.
2. Click Run SETUP.EXE on the AutoPlay menu.
3. When the application prompts you to install the .NET Framework, click OK.
4. Select the Accept radio button, and click Install.
5. When the installation finishes, click Exit.
6. When the application prompts you, click OK to install a hotfix for Windows (KB942288).
7. Click Restart now to restart the server upon completion.
8. After rebooting, log into the server.
9. Click Start, and click Computer.
10. Double-click the CD/DVD drive.
11. Click Installation.
12. Click New SQL Server stand-alone installation.
13. On the Setup support rules screen, click OK.
14. Specify the Enterprise Evaluation free edition. Click Next.
15. Click the checkbox to accept the license terms. Click Next.
16. Click Install to install the setup support files.
17. You may see a Windows Firewall warning. For now, ignore this, and click Next.
18. 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.
19. On the Instance configuration screen, leave the default selection of default instance, and click Next.
20. On the Disk space requirements screen, click Next.
21. 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.
22. On the Database Engine Configuration screen, select Mixed Mode.
23. Enter a password for the system administrator account.
24. Click Add Current user. This may take several seconds.
25. Click Next.
26. On the Error and usage reporting screen, click Next.
27. On the Installation rules screen, click Next.
28. On the Ready to Install screen, click Next.
29. Repeat steps 9 through 28 for the second SQL Server instance (choose named instance on the instance configuration screen).
30. After installation is complete, click Start, Administrative Tools, and Local Security Policy.

31. Double-click Perform Volume Maintenance Tasks, and add the System account.
32. Double-click Lock Pages in Memory, and add the System account.
33. Click Start, All Programs, Microsoft SQL Server 2008, and SQL Server Management Studio.
34. In the Object Explorer pane, choose Connect→Database Engine. Choose the default instance.
35. Right-click the instance, and choose properties.
36. Click Processors, and check the Boost SQL Server Priority checkbox.
37. Repeat steps 34 through 36 on the named instance.
38. Place the prepared DVD Store backup file in the default location for SQL Server: C:\Program Files\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\Backup.

Configuring additional drives in Disk Management

1. Click Start, Administrative Tools, Computer Management.
2. Click Disk Management.
3. Right-click the uninitialized disks (if any), and choose Initialize Disk.
4. Choose MBR.
5. Right-click the volume, and choose New Simple Volume. Click Next.
6. Keep the default drive space, and click Next.
7. Keep the default drive letter assignment, and click Next.
8. Choose NTFS, 64KB allocation unit size, and check the Perform a quick format checkbox. Click Next.
9. Click Finish.
10. Create folders necessary for the database. We used <driveletter>:\SQLData for the SQL Server database volumes and <driveletter>:\SQLLog for the SQL Server transaction log volume.

DVD Store setup

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 2008, and then used that master copy to restore our test database to the Dell PowerEdge R710 solution between each test run. We stored the backup file on the C: drive 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 core. The Intel Xeon Processor E5506 had eight cores, while the Intel Xeon Processor E5520 had 16 logical processors with hyper-threading enabled. We used eight files per filegroup in both cases.
- 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 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.
- 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.
5. Run the batch file C:\Program Files\Microsoft Visual Studio 9.0\Common7\Tools\vsvars32.bat. This sets up the environment variables for us.
3. Execute the following command:

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

Testing procedure

To perform the test, we used a series of batch files. We stored batch files either under the C:\ServerShare folder on the 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 machine, and controller. We use simple file sharing to copy files from machine to machine, as we 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 on each SQL Server instance.
3. Restore the database on each SQL Server instance.
4. Pause briefly to wait for background tasks to complete before the scripts reboot the server.
5. The scripts reboot the client machines and server machine, and wait for a ping response from all test machines.
6. Wait 10 additional minutes for any background tasks to complete.
7. Start the counter logs on client machines, and start the data collector set on the server under test.
8. Start the workload connections.
9. Start the workload ramp-up period.
10. Start the workload on instance number 1. Wait 10 seconds.
11. Start the workload on instance number 2.
12. Stop the workload.
13. Copy all output files to the controller.

Appendix A – Server configuration information

This appendix provides detailed configuration information about each Dell PowerEdge R710 server, which we present in Figure 3.

Servers	Intel Xeon Processor E5506-based Dell PowerEdge R710 server	Intel Xeon Processor E5520-based Dell PowerEdge R710 server
General dimension information		
Height (inches)	3.5	3.5
Width (inches)	17.0	17.5
Depth (inches)	27.0	27.0
U size in server rack (U)	2	2
Power supplies		
Total number	2	2
Wattage of each (W)	870	870
Cooling fans		
Total number	5	5
Dimensions (h x w) of each (inches)	2.5 x 2.5	2.5 x 2.5
Voltage (V)	12	12
Amps (A)	1.60	1.60
General processor setup		
Number of processor packages	2	2
Number of cores per processor package	4	4
Number of hardware threads per core	1	2
System power management policy	OS control	OS control
CPU		
Vendor	Intel	Intel
Name	Xeon E5506	Xeon E5520
Stepping	D0	D0
Socket type	LGA1366	LGA1366
Core frequency (GHz)	2.13	2.26
L1 cache	32 KB + 32 KB (per core)	32 KB + 32 KB (per core)
L2 cache	1 MB (4 x 256 KB)	1 MB (4 x 256 KB)
L3 cache	1 x 4 MB	1 x 8 MB
Platform		
Vendor and model number	Dell PowerEdge R710	Dell PowerEdge R710
Motherboard model number	PWBYN967	PWBYN967
Motherboard revision number	X08	X08
BIOS name and version	Dell 0.2.15	Dell 0.2.15
BIOS settings	Default	Default
Memory modules		
Total RAM in system (GB)	24	24

Servers	Intel Xeon Processor E5506-based Dell PowerEdge R710 server	Intel Xeon Processor E5520-based Dell PowerEdge R710 server
Number of types of memory modules	1	1
Vendor and model number	Crucial CT51272BB1339	Crucial CT51272BB1339
Type	PC3-10600 DDR3	PC3-10600 DDR3
Speed (MHz)	1,333	1,333
Speed in the system currently running @ (MHz)	800	1,066
Timing/latency (tCL-tRCD-iRP-tRASmin)	6-6-6-18	7-7-7-20
Size (GB)	24	24
Number of RAM modules	6 x 4 GB	6 x 4 GB
Chip organization	Double-sided	Double-sided
Hard disk		
Vendor and model number	Seagate ST973451SS	Seagate ST973451SS
Number of disks in system	6	6
Size (GB)	73	73
Buffer size (MB)	16	16
RPM	15,000	15,000
Type	SAS	SAS
Operating system		
Name	Windows Server 2008 Enterprise Edition x64	Windows Server 2008 Enterprise Edition x64
Build number	6001	6001
File system	NTFS	NTFS
Language	English	English
Network card/subsystem		
Vendor and model number	Intel PRO/1000 Dual Port Adapter D33682	Intel PRO/1000 Dual Port Adapter D33682
Type	PCI-E	PCI-E
Optical drive		
Vendor and model number	TEAC DV-28S-VDB	TEAC DV-28S-VDB
USB ports		
Number	4	4
Type	2.0	2.0

Figure 3: Detailed system configuration information for the two test servers.

Appendix B – Storage array configuration information

This appendix provides detailed configuration information about the Dell EqualLogic storage array systems, which we present in Figure 4.

Storage	Dell EqualLogic iSCSI SAN
Arrays	Three Dell EqualLogic PS5000XV arrays
Disks	48 15,000RPM SAS disks total for the three shelves
Active storage cache	1.5 GB (512 MB per array)
Passive storage cache	1.5 GB (512 MB per array)
Number of active storage controllers	3 (1 per array)
Number of passive storage controllers	3 (1 per array)
Number of active storage ports	9 (3 per array)
Number of passive storage ports	9 (3 per array)
Firmware revision	v4.1.1
Switch type/ model/firmware revision	One Dell PowerConnect 6248 v2.2.0.3
Disk vendor and model number	32x Seagate ST3146855SS 16x Maxtor ATLAS15K2_147SAS
Disk size (GB)	146
Disk buffer size (MB)	16
Disk RPM	15,000
Disk type	SAS
RAID type	RAID 10

Figure 4: 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