RACK SERVERS: DATABASE PRICE-PERFORMANCE

Dell™ PowerEdge™ R310
Delivers 5.1% greater price-performance
versus
HP ProLiant DL120 G6
And it offers more expansion room for future growth!

OUR FINDINGS

To minimize data center costs and allow capacity for future growth, companies need servers that deliver both strong price-performance and a variety of expansion options. In Principled Technologies' tests in our labs, the Dell PowerEdge R310 delivered 5.1 percent greater database performance per dollar than the HP ProLiant DL120 G6 server—and it offered more hardware options for future growth. The Dell PowerEdge R310 lets a company spend less to achieve comparable performance today and secure greater options for tomorrow.

OUR PROCESS

To gauge how well each server handles database activity, we used a workload representative of many real-world database applications, DVD Store. This tool is an open-source database benchmark that simulates an online e-commerce store that sells DVDs. Both servers were running Microsoft® Windows Server® 2008 R2 Enterprise Edition with Microsoft SQL Server® 2005 x64.

MAY 2010
A PRINCIPLED TECHNOLOGIES TEST REPORT
Commissioned by Dell Inc.

PROJECT OVERVIEW

The goal of the project was to determine which of the following systems had better database price/performance as well as which server had more advanced hardware options:

- Dell PowerEdge R310
- HP ProLiant DL120 G6

Both servers ran Microsoft Windows Server 2008 R2 Enterprise Edition with Microsoft SQL Server 2005 x64. We used a single client running Microsoft Windows Server 2003 R2 to induce a heavy database load on the server.

We also compared the upgrade capabilities of the online offerings of each server in order to determine which system had superior hardware options.

WORKLOAD

DVD Store Version 2 (DS2) is an accessible, open-source, flexible framework with which we can model a typical online transaction processing (OLTP) database. DS2, which simulates an online e-commerce DVD store, has database components and Web server components, and includes driver programs that put heavy loads on the server. We opted to use the included driver program to stress the database layer directly, and bypassed the Web interface.

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. Each search is by title, actor, or category. The title and actor searches use full-text search. The other customer actions, including adding new customers, exercise a wide range of database functions.

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 machines. While the DVD Store client application outputs OPM at 10-second intervals visually, we chose to collect this OPM metric via the performance monitor counter on each client at 1-second intervals. We ran this workload on each server for 35 minutes, which consisted of 5 minutes of warm-up time and 30 minutes of run time.

Each server under test ran a single Microsoft SQL Server 2005 x64 instance with a 5GB database that we tuned to saturate the server's processor and memory. We used the default DS2 parameters and setup configuration, with the exceptions we note in the DVD Store setup section in the How we tested section of this report.

Each client machine ran a single instance of DS2, with 24 threads, which simulate users executing actions within the application. This simulated a heavily loaded environment; the load-generating client machine ran with no think time, blasting requests as quickly as the servers could handle them. For more details on the DS2 tool, see http://www.delltechcenter.com/page/DVD+Store.

SYSTEM COMPARISON

Figure 1 highlights some similarities and differences between the two servers. Appendix A presents detailed system information.

Hardware specifications	Dell PowerEdge R310	HP ProLiant DL120 G6
CPU	Intel® Xeon® X3450	Intel Xeon X3450
CPU speed (GHz)	2.66	2.66
Number of processor packages	1	1
Number of cores per processor package	4	4
Number of hardware threads per core	2	2
Memory type	PC3-8500R	PC3-8500R
Total memory (GB)	8	8
Maximum supported memory (GB)	32	24
Redundant power supply	Available	Not available

Figure 1: System configuration information for the test servers.

As Figure 1 shows, the servers are very similar. The primary differences lie in the Dell PowerEdge R310's greater potential for future expansion in the following two areas:

- **Redundant power supply.** The Dell PowerEdge R310 is available with a redundant power supply option, while the HP ProLiant DL120 G6 only offers non-redundant configurations. Redundant power supplies prevent server downtime in the event of a power supply failure.
- **Maximum memory.** The Dell PowerEdge R310 offers expansive memory configuration options supporting up to 32GB RAM, while the HP ProLiant DL120 G6 supports only 16GB RAM.

Note: We configured our test systems with 8 GB of total RAM using two 4GB PC3-8500R, 1,066MHz, quad-ranked memory modules. You can alternatively configure the Dell PowerEdge R310 for faster memory performance – also at 8GB total RAM – using four 2GB PC3-10600R, 1,333MHz, dual-ranked memory modules.

WHAT WE FOUND

As Figure 2 shows, the Dell PowerEdge R310 achieved a DVD Store Version 2 price-performance score of 18.93 OPM/dollar, a 5.1 percent greater price-performance score than the HP ProLiant DL120 G6, which achieved a price-performance score of 18.00 OPM/dollar. To calculate the price-performance score, we divided OPM by price.

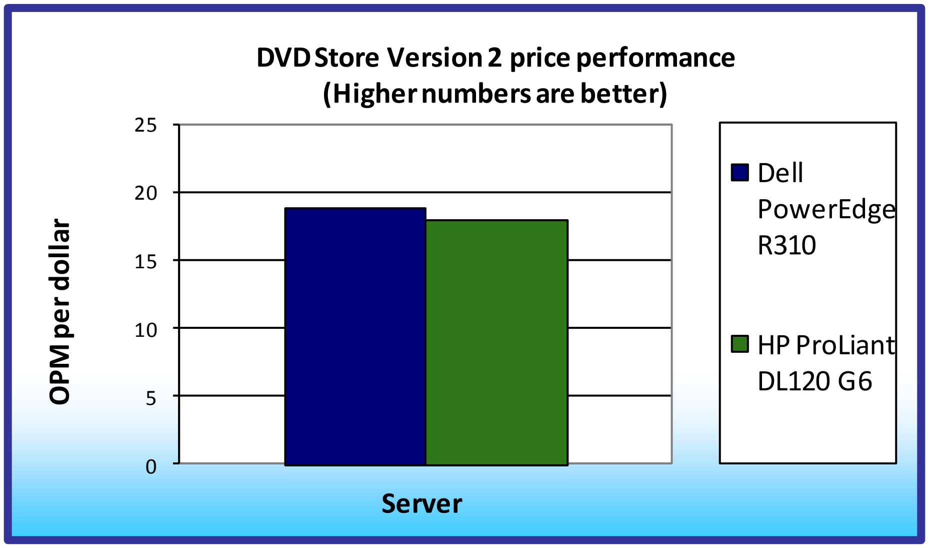

Figure 2: Normalized DVD Store Version 2 price-performance results for the two servers. Higher numbers are better.

Figure 3 shows the prices of the two servers on April 27, 2010. The Dell PowerEdge R310, at $4,689, cost 3.3 percent less than the HP ProLiant ML110 G6 at $4,844.

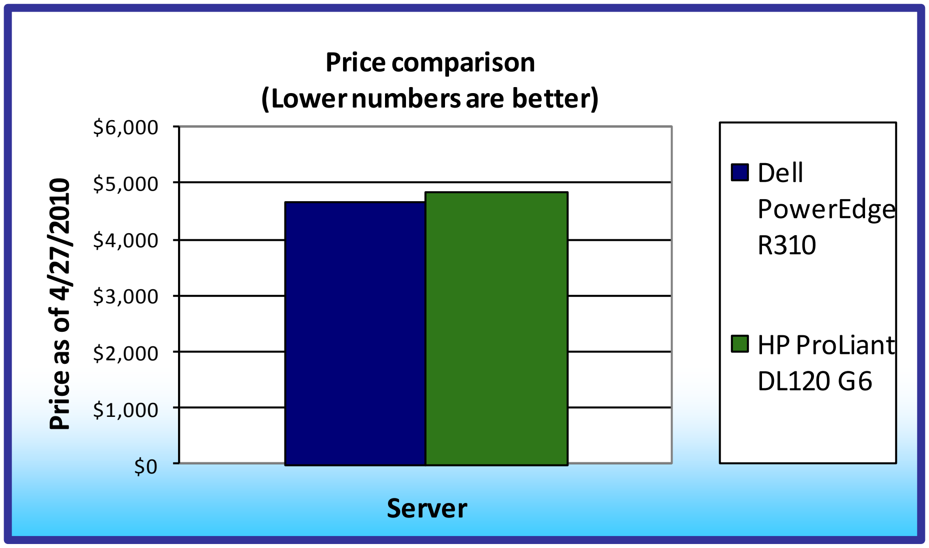

Figure 3: Price comparison for the two servers. Lower numbers are better.

As Figure 4 shows, the DVD Store Version 2 performance of the two servers is almost identical: the Dell PowerEdge R310 achieved a DVD Store Version 2 score of 88,746 OPM, 1.8 percent greater than the HP ProLiant DL120 G6's score of 87,211 OPM.

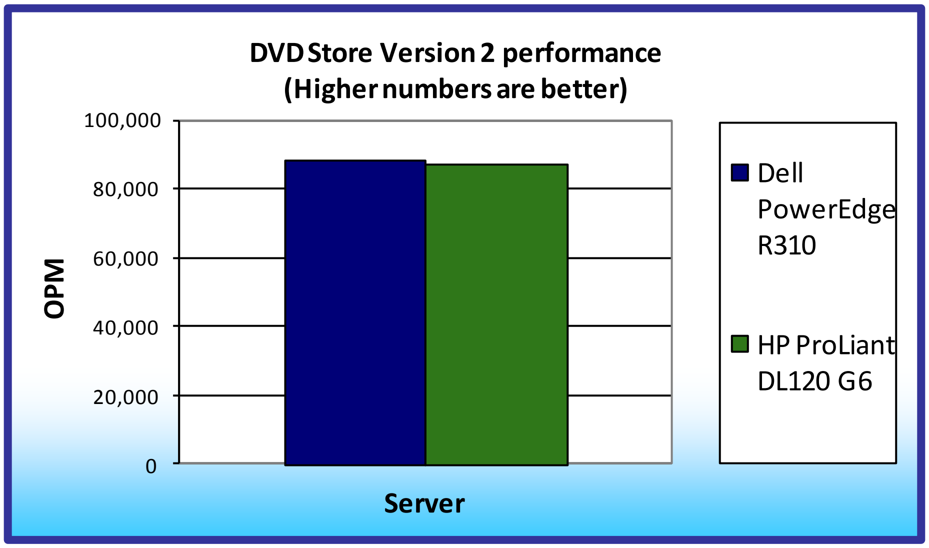

Figure 4: DVD Store Version 2 performance results, in OPM, for the two servers. Higher numbers are better.

HOW WE TESTED

Setting up the servers

For each server, we configured four internal SAS hard drives in two RAID 1 volumes, one for the operating system and the other for the SQL Server database files. We installed a fresh copy of Microsoft Windows Server 2008 R2 Enterprise Edition on both servers.

Installing Microsoft Windows Server 2008 R2 Enterprise Edition on the host server

1. Boot the server, and insert the Windows Server 2008 R2 installation DVD in the DVD-ROM drive.
2. At the Language Selection Screen, click Next.
3. Click Install Now.
4. Select Windows Server 2008 R2 Enterprise (Full Installation), 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. Ensure you select the proper drive, and 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 the network configuration on the server

1. Click Start→Control Panel→Network and Internet→Network Connections, and double-click the Local Area Connection assigned to client/server 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. Click OK, and click Close to exit.

Installing system updates in Microsoft Windows Server 2008 R2

We installed the following updates on each server using the Windows Update feature:

- Security Update for Windows Server 2008 R2 x64 Edition (KB972270)
- Security Update for Windows Server 2008 R2 x64 Edition (KB974571)
- Cumulative Security Update for Internet Explorer 8 for Windows Server 2008 R2 x64 Edition (KB978207)
- Windows Malicious Software Removal Tool x64 - January 2010 (KB890830)
- Update for Windows Server 2008 R2 x64 Edition (KB974431)
- Update for Windows Server 2008 R2 x64 Edition (KB976098)
- Security Update for ActiveX Killbits for Windows Server 2008 R2 x64 Edition (KB973525)
- Security Update for Windows Server 2008 R2 x64 Edition (KB975467)

Installing Microsoft SQL Server 2005 x64

1. Insert the installation DVD for SQL Server 2005 x64 into the DVD drive.
2. Navigate to the SQL Server 2005 x64 DVD, and double-click. Choose the appropriate edition if the installer prompts you with a menu.
3. Click the Server components link under the Install section 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 may 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, select 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. To complete the installation, click Finish.

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 (see the Setting up DVD Store section below for the specific script to use).
25. Restart the server before installing SQL Server 2005 x64 SP3.

Installing Service Pack 3 for Microsoft SQL Server 2005 x64

Service Pack 3 is available for download from http://www.microsoft.com/downloads/details.aspx?FamilyID=ae7387c3-348c-4faa-8ae5-949fdfbe59c4&displaylang=en

1. Run SQLServer2005SP3-KB955706-x86-ENU.exe.
2. At the Welcome window, click Next.
3. At the License Terms window, select I accept the agreement, and click Next.
4. At the Feature Selection window, accept default selections, and click Next.
5. At the Authentication window, click Next.
6. At the Error and Usage Reporting Settings window, click Next.
7. At the Running Processes window, click Next.
8. Click Install.
9. If a message to reboot the computer appears, click OK.
10. At the Installation Progress window, click Next.
11. At the Installation Complete window, click Next.
12. Uncheck the Launch the User Provisioning Tool for Windows Vista after SP3 installation completes option, and click Finish.
13. Reboot the server.

Installing and configuring the database clients

For the DS2 scripts, we used a client system to simulate a number of users putting a load on the server. On the client machine, we created a folder we called C:\ClientShare to store workload scripts. We installed the .NET 3.5 framework on each client, as the DS2 test executable requires at least .NET2.0. We created a performance counter log 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 of `Clientx` for the database client, where x is the client number.
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.
8. Install Windows Updates, .NET 3.5 framework, and copy the DVD Store client executable into the c:\clientshare folder.

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 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, and triggers. We built the 5GB database in SQL Server 2005 and then performed a full backup, storing the backup file on the C: drive for quick access. We used that backup file to restore both servers between test runs.

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. 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 the default 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. Both servers under test have four cores. We used four files per filegroup in all 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 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 - 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
```

Testing procedure

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

The testing procedure consisted of the following steps:

1. Reboot the server and the client machine.
2. Wait 5 minutes for any background tasks to complete.
3. Begin recording idle power for two minutes.
4. Start the workload connections.
5. Start the workload ramp up period.
6. Start the workload.
7. Stop the workload.
8. Copy all output files.
9. Drop the database on each instance.
10. Restore the database on each instance.

APPENDIX A – SERVER CONFIGURATION INFORMATION

Figure 5 provides detailed configuration information about the test servers.

Servers	Dell PowerEdge R310	HP ProLiant DL120 G6
General processor setup		
Number of processor packages	1	1
Number of cores per processor package	4	4
Number of hardware threads per core	2	2
CPU		
Vendor	Intel	Intel
Name	Xeon X3450	Xeon X3450
Stepping	B1	B1
Socket type	LGA1156	LGA1156
Core frequency (GHz)	2.66	2.66
Bus frequency	2.5 GT/s	2.5 GT/s
L1 cache	32 KB + 32 KB (per core)	32 KB + 32 KB (per core)
L2 cache	256 KB (per core)	256 KB (per core)
L3 cache (MB)	8	8
Thermal design power (TDP, in watts)	95	95
Platform		
Vendor and model number	Dell PowerEdge R310	HP ProLiant DL120 G6
Motherboard chipset	Intel 3420	Intel 3420
BIOS name and version	Dell 1.0.0 (04/13/2010)	HP 0260126 (01/26/2010)
BIOS settings	Default	Default
Memory modules		
Vendor and model number	Hynix HMT151R7BFR8C-G7	Micron MT36JSZF51272PDZ-1G1F1AB
Type	PC3-8500R DDR3	PC3-8500R DDR3
Speed (MHz)	1,066	1,066
Speed in the system currently running @ (MHz)	1,066	1,066
Timing/Latency (tCL-tRCD-iRP-tRASmin)	7-7-7-20	6-6-6-16
Size (GB)	8	8
Number of RAM modules	2 x 4 GB	2 x 4 GB
Chip organization	Double-sided	Double-sided

Servers	**Dell PowerEdge R310**	**HP ProLiant DL120 G6**
Hard disk		
Vendor and model number	Dell ST3300657SS	HP EF0300FARMU
Number of disks in system	4	4
Size (GB)	300	300
Buffer size (MB)	16	16
RPM	15,000	15,000
Type	SAS 6Gb/s	SAS 6Gb/s
Controller	Dell PowerEdge RAID Controller (PERC) H700	HP Smart Array P212 Controller
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 BCM5716C NetXtreme II Gigabit Ethernet	Broadcom BCM5723 NetXtreme Gigabit Ethernet
Type	Integrated	Integrated
USB ports		
Number	4	6
Type	USB 2.0	USB 2.0

Figure 5: 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.

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.