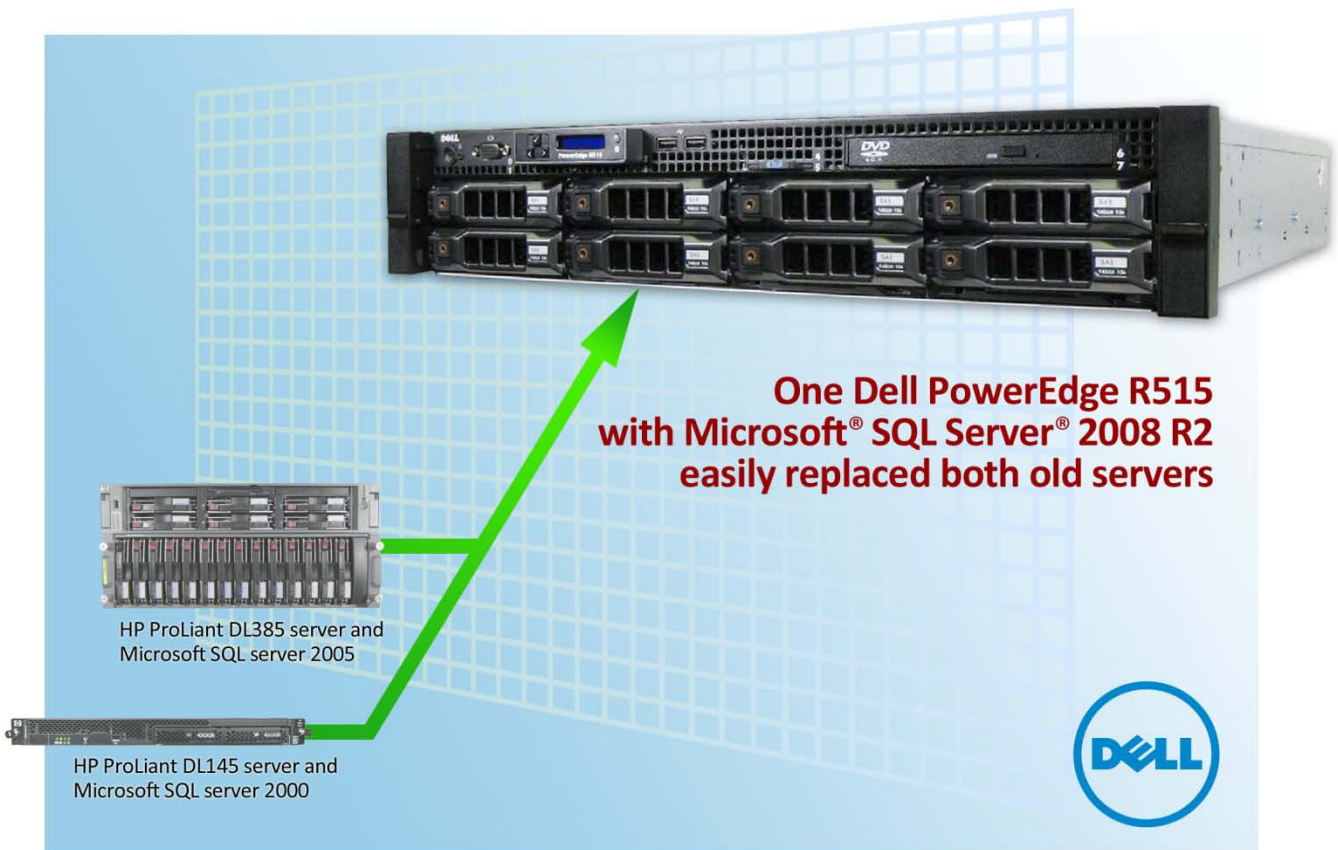



# Migrating to a Dell™ PowerEdge™ R515 from two legacy database servers



**One Dell PowerEdge R515 with Microsoft® SQL Server® 2008 R2 easily replaced both old servers**

HP ProLiant DL385 server and Microsoft SQL server 2005

HP ProLiant DL145 server and Microsoft SQL server 2000



**One Dell PowerEdge R515 handled the tough demands of a growing business AND provided plenty of room for future growth**

# TABLE OF CONTENTS

Table of contents .....	2
Executive summary .....	3
Why consider migrating to SQL Server 2008 R2 on a Dell PowerEdge R515? .....	4
Features of the new Dell PowerEdge R515 .....	4
Features of Microsoft Windows Server 2008 R2 and Microsoft SQL Server 2008 R2 .....	5
Consolidation with Dell technology simplifies and adds features.....	6
The power of consolidation .....	6
We show you how: Preparing for the move.....	9
Evaluating your databases .....	9
Configuring your internal storage.....	10
Installing Windows Server 2008 R2 and enabling the Hyper-V role .....	11
Creating your virtual machines.....	12
Installing SQL Server 2008 R2 on your virtual machines .....	13
We show you how: Making the move .....	16
Upgrade Advisor makes it easy.....	16
Side-by-side migration .....	19
We show you how: After the move .....	23
Logins and dependencies.....	23
Summing it all up .....	26
Appendix A – Example database survey .....	27
Appendix B – Preparing the storage .....	29
Appendix C – Installing windows and enabling Hyper-V .....	30
Appendix D – Creating virtual machines.....	32
Appendix E – Installing SQL Server 2008 R2 .....	34
Appendix F – Installing Upgrade Advisor .....	36
Appendix G – Migrating databases.....	38
Backing up a SQL Server 2000 database .....	38
Backing up a SQL Server 2005 database.....	39
Restoring a database to SQL Server 2008 R2.....	40
Appendix H – Transferring Windows logins.....	42
Appendix I – Transferring SQL Server logins.....	43
Mapping transferred logins to database users.....	43
About Principled Technologies .....	45

## EXECUTIVE SUMMARY

Our hypothetical SMB replacing multiple older HP servers with a single Dell PowerEdge R515 server running Microsoft SQL Server 2008 R2 benefits in several ways:

- Significant performance increase
- Total cost of investment (TCO) and return on investment (ROI) within 19 months
- Easy migration process

This report addresses the ease of the migration process. To learn more about performance increase, see our [performance report](#). To learn more about the TCO and ROI, see our [TCO/ROI report](#).

Small to medium businesses (SMBs) with legacy database servers, such as older HP ProLiant systems, often find themselves running out of capacity and at a decision point: Do they continue with the status quo, almost guaranteeing increased hardware failures and lost revenue and time, or do they migrate to newer and more powerful servers? With the release of the Dell PowerEdge R515 and the power of Microsoft's Hyper-V™ and SQL Server technologies, the choice is easy for the SMB market: upgrading to the Dell PowerEdge R515 is simple, cost-effective, and results in better performance with more room for growth.

This Principled Technologies® (PT) Guide explains the process for consolidating an existing two-server installation onto a single Dell PowerEdge R515 (see Figure 1) and upgrading a legacy heterogeneous SQL Server 2000 and SQL Server 2005 environment to a unified Microsoft SQL Server 2008 R2 platform. We provide concepts and procedures that will help you successfully consolidate database instances from multiple older servers onto a single Microsoft Windows Server® 2008 R2 system running Hyper-V, with virtual machines (VMs) running SQL Server 2008 R2.



**Figure 1. The new Dell PowerEdge R515 server.**

In the sections that follow, we discuss some of the new features of the servers and software, give an overview of the consolidation and

migration process, and discuss post-migration considerations. The appendices to the Guide give detailed, step-by-step information on how to do the migration.



**NOTE:** Those with more complex configurations, special requirements, or the desire to avoid doing the migration themselves may wish to investigate the consulting services Microsoft and Dell offer.

## Why consider migrating to SQL Server 2008 R2 on a Dell PowerEdge R515?

Because any migration requires an investment of time, it should deliver clear benefits. In this section, we present just a few of the benefits of migrating to SQL Server 2008 R2 on a Dell PowerEdge R515.

## Features of the new Dell PowerEdge R515

The Dell PowerEdge R515 offers many new features for maximizing performance and for minimizing operational expenses, including the following:

- **Power.** The Dell PowerEdge R515 includes enhancements that let it use less energy than many older servers do. When you consolidate multiple legacy servers onto a single Dell PowerEdge R515, the potential power savings are dramatic. We highlight this cost savings in our accompanying TCO/ROI report.<sup>1</sup>
- **Processors.** The Dell PowerEdge R515 uses the AMD Opteron™ 4100 Series processors. This makes the Dell PowerEdge R515 an excellent yet affordable virtualized database consolidation platform.
- **Management.** The Dell PowerEdge R515, like all late-model Dell servers, comes with the Dell Lifecycle Controller. This tool simplifies server management by providing a single interface for management functions and by storing critical system

---

<sup>1</sup>“Dell PowerEdge R515 offers savings over two legacy database servers”  
[http://principledtechnologies.com/clients/reports/Dell/R515\\_TCO\\_ROI.pdf](http://principledtechnologies.com/clients/reports/Dell/R515_TCO_ROI.pdf)

information in the system itself. There are no CDs or USB keys to keep track of for drivers or firmware.

- **Storage.** The Dell PowerEdge R515, available in both an 8-bay and 12-bay chassis, holds up to 25 terabytes of internal storage, giving you many times the capacity of prior common small business servers.
- **Memory.** The Dell PowerEdge R515 holds up to 64 GB<sup>2</sup> of RAM, many times the RAM amounts of older small business servers, allowing for powerful flexibility with virtualized database solutions.

## Features of Microsoft Windows Server 2008 R2 and Microsoft SQL Server 2008 R2

An SMB is not necessarily a simple business. Business needs, regulatory requirements, and other factors place an increasing strain on IT resources that are frequently quite limited. Microsoft Windows Server 2008 R2 and SQL Server 2008 R2 address many of these issues. For more information about Microsoft Windows Server 2008 R2, see <http://www.microsoft.com/windowsserver2008/en/us/default.aspx>. For more information about Microsoft SQL Server 2008 R2, see <http://www.microsoft.com/sqlserver/2008/en/us/r2.aspx>. Below, we highlight a few of the major features of this new software platform.

- **Virtualization.** Microsoft Windows Server 2008 R2 includes Microsoft Hyper-V, which you can use to create virtual machines. These virtual servers, which are often more powerful than your previous physical machines, can host your critical business applications and databases. Virtual machines also allow you to isolate business functions for security or regulatory reasons.
- **Simplified installation.** The wizard-based setups of Windows Server 2008 R2 and SQL Server 2008 R2 make the installation and configuration process considerably simpler than in past versions.
- **Windows Management and updated components.** Microsoft Windows Server 2008 R2 includes a revamped Server Manager console that contains almost all server management functions, including the ability to quickly add Roles and Features to your server.

---

<sup>2</sup>The Dell PowerEdge R515 will support 128 GB of RAM by the end of 2010.

- **SQL Server Management and updated components.** SQL Server 2008 R2 is the latest release of Microsoft’s database management platform. As with each release, Microsoft has added new features to expand on the capabilities of their Database Management System (DBMS) platform. Where earlier versions of SQL Server required database administrators to rely primarily on either in-house applications or third-party tools to monitor multiple instances, SQL Server 2008 R2 simplifies multi-server management. The Standard Edition includes many of the same business intelligence (BI) subsystems that larger enterprise installations need. Finally, SQL Server 2008 R2 contains many new management features, such as Policy Based Management and PowerShell Support.

## CONSOLIDATION WITH DELL TECHNOLOGY SIMPLIFIES AND ADDS FEATURES

Using AMD Opteron 4100 Series processors and Microsoft Hyper-V technology, the Dell PowerEdge R515 allows organizations to consolidate via virtualization while adding features, creating additional performance headroom, and realizing savings all at once.

### The power of consolidation

#### What is consolidation?

Generally speaking, *consolidation* is the process of combining multiple items to make a single, more effective unit. In an IT context, you can consolidate the following:

- **Physical servers.** After a successful server consolidation, all applications should run on fewer servers than before. Ideally, those applications should run at least as well as they did previously, and potentially better. In our accompanying performance report,<sup>3</sup> we highlight the dramatic performance gains realized in a sample migration.
- **Storage.** Depending on your setup, consolidating servers may let you also consolidate storage by moving data from a

---

<sup>3</sup>“Dell PowerEdge R515 outperformed two legacy database servers”

[http://principledtechnologies.com/clients/reports/Dell/R515\\_performance.pdf](http://principledtechnologies.com/clients/reports/Dell/R515_performance.pdf)

number of servers to a single large disk storage subsystem in a new server. In the case of the Dell PowerEdge R515, this is possible due to its internal storage capacity- up to 16TB for the 8 hard drive chassis or up to 25TB for the 12 hard drive chassis.

- **Space.** As you consolidate servers, you will likely reduce the number of racks or even the number of locations that house servers.

In this database-specific Guide, we address the consolidation of a heterogeneous environment containing SQL Server 2000 and SQL Server 2005 instances to a Dell PowerEdge R515 running Windows Server 2008 R2 with Hyper-V, with SQL Server 2008 R2 inside virtual machines. As we will demonstrate, consolidating multiple legacy database instances to one physical machine is a simple and easy-to-follow process.

### Why consolidate?

An effective server consolidation effort has the potential to yield an environment with more consistent management practices and improved reliability, security, and hardware utilization—all while maintaining or exceeding the previous level of application performance.

Consolidation can also yield a variety of cost savings:

- **Hardware savings.** Buying, powering, and supporting fewer servers brings obvious savings. Other potential hardware cost savings include the need for fewer racks and network switches: as the number of servers decreases, these costs decrease as well.
- **Maintenance and staff savings.** A consolidated infrastructure offers many opportunities to save on maintenance, support, and staffing. Less hardware and associated equipment means fewer servers that require security patches, monitoring, and other ongoing maintenance.
- **Reduced support costs.** The cost of a given level of support is typically proportional to the size of the installation. As the

number of servers decreases, support costs are also likely to decrease.

- **Power and cooling savings.** Consolidating servers saves power and cooling by using fewer, more efficient systems.

### Sizing and baseline performance

One key to a successful consolidation is sizing, the process of gathering different performance baselines so you have an approximate set of requirements the new hardware platform must meet. You do this by determining the performance characteristics of existing hardware during normal business operations, and then applying growth and scalability estimates.

Among the potential characteristics to examine on each server are the following:

- Processor utilization
- Typical and heavy load memory requirements of the operating system and applications
- Disk layout
- Database size
- Expected database growth
- Maximum concurrent users
- Types and rates of transactions against the databases

The server you select for consolidation must do more than match the combined capacity of your current servers. It also must have enough excess capacity to still perform well at the end of its expected life span.

The Dell PowerEdge R515 we tested for this Guide is a high-performance server focused on the small to medium business market segment that can readily support significant consolidation. It has two AMD Opteron 4100 Series processors, can support up to 64 GB<sup>4</sup> of

---

<sup>4</sup>The Dell PowerEdge R515 will support 128 GB of RAM by the end of 2010.



RAM, and has been optimized to reduce both power consumption and heat dissipation.

## WE SHOW YOU HOW: PREPARING FOR THE MOVE

Our accompanying performance and TCO/ROI reports show from performance and cost perspectives why you should consolidate your older database servers to a single Dell PowerEdge R515 and use Hyper-V virtualization technologies with the added benefits of a newer SQL Server 2008 R2 release. In this Guide, we address the next question: How to accomplish this? We discuss planning issues, setting up the Dell PowerEdge R515, the actual migration using a simple backup/restore method, and post-migration considerations.

### Evaluating your databases

As with any migration or consolidation, planning is a key element. This is especially true when moving to a virtualized environment. You must be aware of many specific details related to each physical server you target for consolidation, including the maintenance window in which you will migrate the server to its new environment, the users the move will affect, and the configuration tasks necessary to assimilate the databases into your consolidated environment. A more comprehensive example survey appears in [Appendix A](#). Information to gather before consolidation includes the following:

- Server OS version and patch level
- SQL Server version and patch level
- Number of logins on this SQL Server instance, and what type of logins these are (Windows or SQL)
- Current backup strategy and schedule for the databases on this server
- Replication details for this SQL instance, if any
- Detailed information regarding permissions and roles
- SQL Server Agent jobs on this SQL Server

After moving your databases to their new SQL Server instance, you must make sure that any system or application using the database has updated connection information. This includes logins, permissions, applications, SQL Server Agent jobs, third-party backup products, and so on.

### **Gathering baseline performance data**

During your research phase, you should use Windows® Performance Monitor, SQL Server Profiler, and other tools to gather data on the typical query load and performance statistics on the databases you are considering moving to the new environment. This effort serves two purposes. First, it provides a prime opportunity to identify potential problems before you move to a consolidated solution. Second, you can use the information you gather to map out your resource allocation needs, which you can then use to configure your new virtual machines, either by using different memory allocations, or by allocating differing numbers of virtual CPUs on the virtual machines.

## **Configuring your internal storage**

In our test bed, we used only internal storage on the Dell PowerEdge R515, as this adequately served our scenario and our upgrade path. The Dell PowerEdge R515 offers up to 25TB of internal storage, which is likely a huge storage capacity boost for many small businesses. This section provides an overview of the setup of the internal storage configuration on the Dell PowerEdge R515. [Appendix B](#) provides complete, detailed installation instructions.

- 1.** Power on the Dell PowerEdge R515, and press Ctrl+R to enter the PERC BIOS Configuration Utility when the system prompts you during boot. Select the internal RAID controller if offered the choice.

2. Highlight the controller, and press F2 to view your options. Choose to create a new virtual disk.
3. Select the relevant disks and the RAID level required for your scenario. Use the Tab key to navigate through the available options. Highlight OK, and press Enter to complete the virtual disk creation process.
4. We recommend completely initializing the new volume after creation. Highlight the volume by choosing F2, choose Full initialization, and begin the initialization.
5. After the installation completes, press Escape to exit the PERC BIOS Configuration Utility, and press Ctrl+Alt+Delete to reboot the system.

## Installing Windows Server 2008 R2 and enabling the Hyper-V role

After you configure the internal storage on the Dell PowerEdge R515, you must install the operating system, in this case, Windows Server 2008 R2. We installed Windows Server 2008 R2 with all the default options, as this provides a robust out-of-the-box solution for most SMBs. With the new and improved Server Manager feature, you can easily add Roles and Features to your installation after the initial configuration, if necessary. Below, we provide an overview of the steps required to install the operating system. For detailed steps, see [Appendix C](#).

1. Insert the Windows Server 2008 R2 DVD media into the DVD drive of the Dell PowerEdge R515 server.
2. By default, the server will boot to the DVD drive. If not, you can force the server to boot to the DVD drive by selecting F11 for the one-time boot option prompt.
3. Choose your language, select your edition, select your drive and partition options, and click Next to complete the installation (see Figure 2). The server will reboot several times during the installation process.

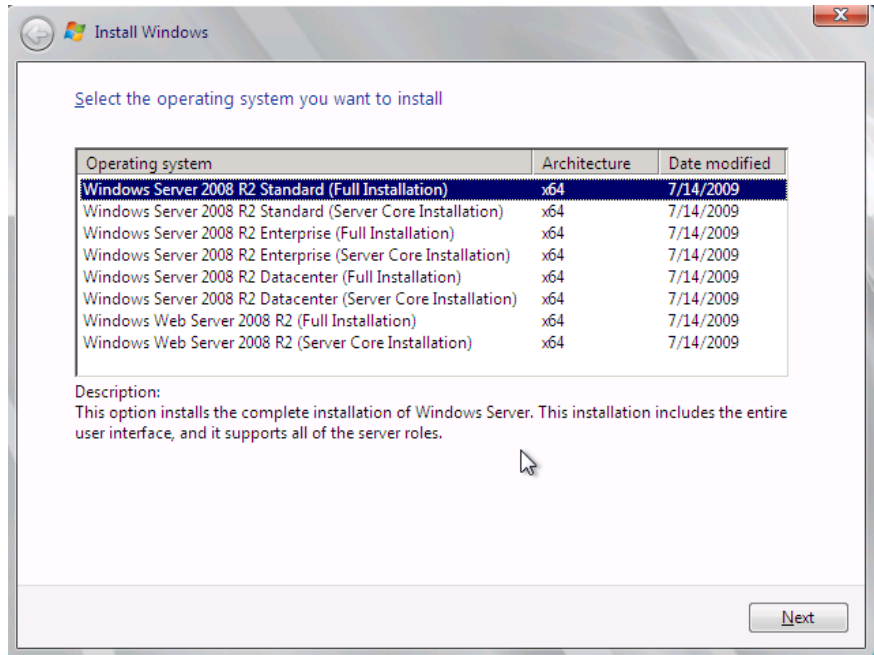


Figure 2. Choosing your edition in the Windows Server 2008 R2 installation.

4. After the operating system installation completes, start the Server Manager.
5. Choose to add the Hyper-V role, and allow the installation process to continue. The server will reboot several times.

## Creating your virtual machines

To complete your preparation for your SQL Server installation in your new virtualized environment, you must finally create virtual machines on which to run the database software. Here, we provide an overview of the steps required to create your first virtual machine using Hyper-V Manager. For detailed steps on this process, see [Appendix D](#).

1. Click Start → Administrative Tools, and click Hyper-V Manager.
2. Create a new VM, assigning two virtual processors, 6GB of RAM, and create a 25GB VHD for the operating system to be installed on. Create additional VHD files if necessary. You may vary your resource allocations, such as virtual processors and RAM amounts, based on your specific needs. See Figure 3 for an example of the VM properties screen.
3. Insert the Windows Server 2008 R2 installation media, and attach the physical media to the VM.

4. Start the VM and install Windows Server 2008 R2, which will require several reboots of the VM.

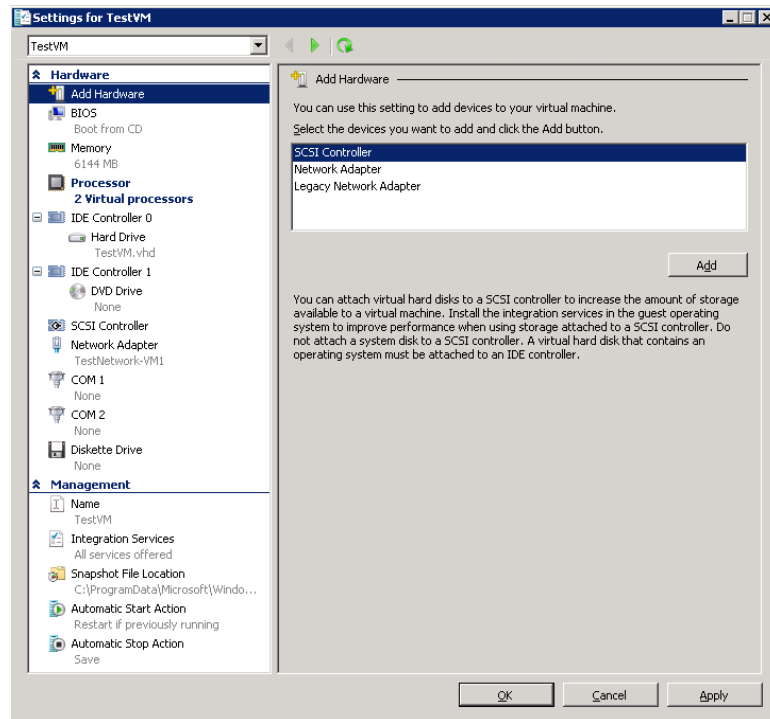


Figure 3. Virtual Machine configuration in Hyper-V.

## Installing SQL Server 2008 R2 on your virtual machines

After you create your virtual machines, you must install SQL Server 2008 R2 on each virtual machine. Although not required, you should, when possible, deploy SQL Server 2008 R2 on machines that are part of an Active Directory® domain. If you do choose to deploy SQL Server in a domain, do not make the SQL Server service domain accounts members of the Domain Administrators group. In fact, grant only the necessary rights on the local server to the SQL Server service account as part of your pre-installation planning. The SQL Server installation software creates the local groups it needs for its security purposes.

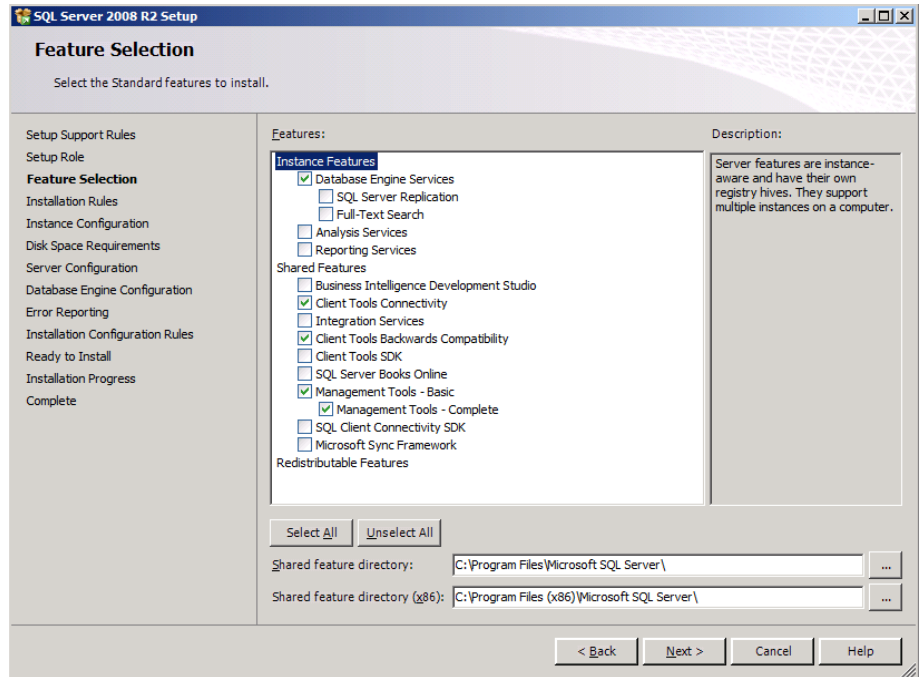
This section provides an overview of the SQL Server 2008 R2 installation process. [Appendix E](#) provides detailed installation instructions.

1. Insert the SQL Server 2008 R2 DVD into the host machine DVD drive, mount it to the virtual machine, and log into the virtual machine. If prompted to enable the .NET Framework Core role, click OK.
2. On the Installation Center screens, choose Installation, and choose to proceed with a new installation. (See Figure 4.)



**Figure 4. SQL Server 2008 R2 Installation Center options.**

3. Proceed through the first several installation steps, entering license information, and installing prerequisites with default options specified. On the Setup Role screen, choose a SQL Server Feature Installation.
4. On the Feature Selection screen, select only what you need for your particular configuration (see Figure 5). In our case, we chose the Database Engine Services and Management Tools.



**Figure 5. Choosing features for installation.**

5. On the first installation, choose to install the default instance. If you perform subsequent installations on the same server, choose named instance and provide a name for the instance.
6. Configure the credentials of the SQL Server service account and SQL Server Agent accounts.
7. Specify SQL Server administrators, and specify the authentication mode desired for your configuration. Microsoft recommends Windows Authentication mode, but legacy applications may require SQL Server authentication.
8. Complete the installation.
9. Repeat this process for any remaining virtual machines on this host.

## WE SHOW YOU HOW: MAKING THE MOVE

### Upgrade Advisor makes it easy

The SQL Server 2008 R2 Upgrade Advisor is a major aid in migration research. This utility, which the SQL Server 2008 R2 setup wizard includes, scans legacy databases and SQL Server components for compatibility issues, features, and syntax the newer DBMS does not support, as well as many other critical components. The utility lets you view reports quickly in the Upgrade Advisor interface or save reports for later review.

You can install and execute the Upgrade Advisor on machines running Windows XP SP2 or greater, Windows Vista® SP1, Windows 7, Windows Server 2003 SP2, Windows Server 2008 SP2, or Windows Server 2008 R2. The Microsoft .NET™ framework is also a requirement.



**BEST PRACTICE:** Use the Upgrade Advisor tool on your SQL Server 2000 and SQL Server 2005 databases and import a trace file to the Upgrade Advisor tool for analysis. The trace file lets the Upgrade Advisor detect issues that might not show up in a simple scan of the database, such as TSQL embedded in applications. Your migration research and planning must account for such instances. You can capture traces of TSQL using SQL Server Profiler on your SQL Server 2000 and SQL Server 2005 servers during typical hours and analyze these traces using the Upgrade Advisor.

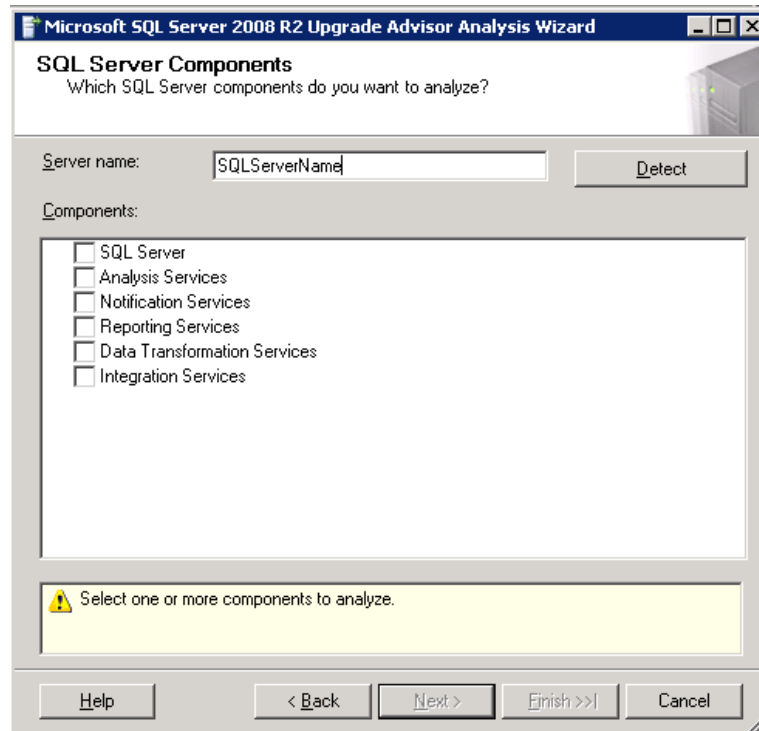
To install the Upgrade Advisor, use the following steps:

1. Insert the SQL Server 2008 R2 DVD. Click the Planning link, and click Install SQL Server Upgrade Advisor.
2. Click Next to begin the installation wizard, accept the licensing terms, and click Next.
3. Click Next to accept the default Registration information, click Next to accept the default installation path, and click Next to begin the installation.



Once you have installed the SQL Server 2008 R2 Upgrade Advisor, you can use this software to scan your SQL Server 2000 and SQL Server 2005 instances for potential migration issues. This section provides a brief walkthrough; [Appendix F](#) gives more detailed instructions.

4. Select Start→All Programs→Microsoft SQL Server 2008 R2→SQL Server 2008 R2 Upgrade Advisor.
5. Click the Launch Upgrade Advisor Analysis Wizard link, and click Next to begin.
6. Enter the older SQL Server computer name, and select the features you want the Upgrade Advisor to analyze. Alternatively, click Detect to have the Upgrade Advisor remotely scan the older SQL Server machine and detect which components are running on the server. (See Figure 6.)



**Figure 6. Upgrade Advisor component selection.**

7. Choose your instance, provide proper authentication to the older SQL Server, and select the database you want to analyze. If necessary, you can also input SQL script files and trace files here.
8. You may have legacy Data Transformation Services (DTS) packages on SQL Server 2000 servers. The DTS packages may be present in the file system or in the database itself. If you selected DTS, or if the Upgrade Advisor automatically detected the presence of DTS

packages, you must now select the DTS location you want to analyze. Click Run to start the analysis. Run times vary.

9. Following the analysis of your older SQL Server instance, you can view the Upgrade Advisor report, which lists warnings and errors. Each warning or error will have associated information to solve the issue. (See Figure 7.)

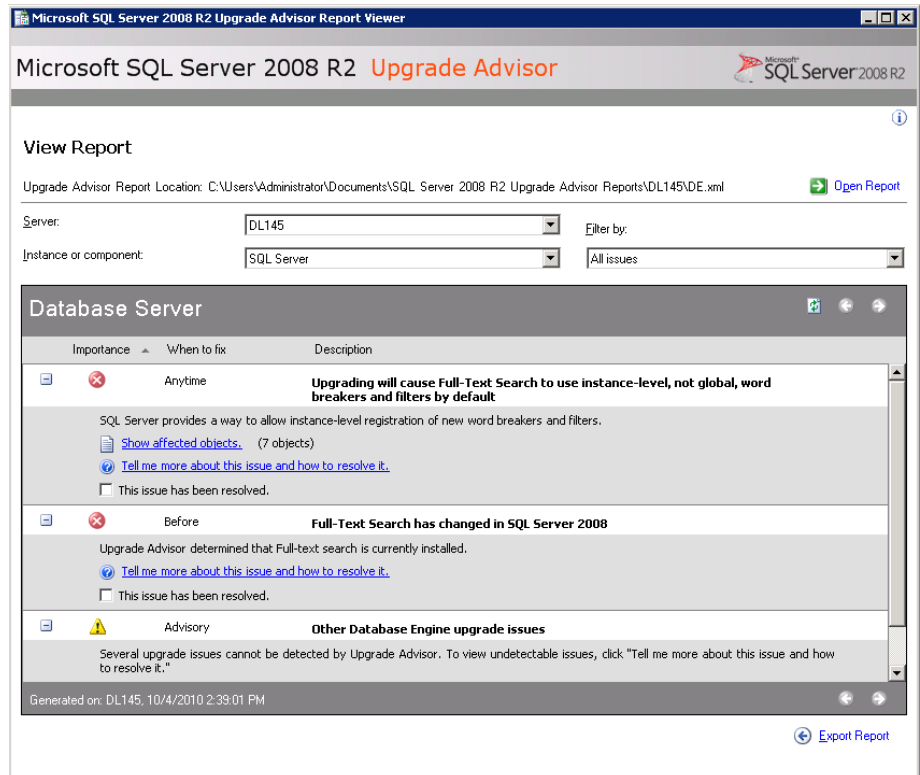


Figure 7. Sample Upgrade Advisor output report.

## Side-by-side migration

In this section, we provide an overview of the processes involved in migrating your database from either SQL Server 2000 or SQL Server 2005 to a virtualized SQL Server 2008 R2 environment on the Dell PowerEdge R515. We provide detailed instructions in [Appendix G](#). We performed all SQL Server 2000 administration using Query Analyzer and Enterprise Manager, the two main tools in the SQL Server 2000 environment. Likewise, we performed all administration for the SQL Server 2005 and SQL Server 2008 R2 installations using SQL Server Management Studio, the main administration interface for those versions.



**NOTE:** Various methods exist for migrating databases between instances and versions of SQL Server, including detach/attach, backup/restore, and the copy database wizard. We chose to use the backup/restore method because most businesses will have a backup routine in place, and administrators will be familiar with the process.

### **Moving your SQL Server 2000 database: performing the backup**

Using the SQL Server 2000 server, take the following steps to back up your database. Backup times vary. Note that you should do any special planning in advance to minimize your downtime window.

1. To keep users from issuing updates during the migration process, you can either set the database to read-only mode or set the access property to SINGLE\_USER. Both choices immediately sever all user connections. See [Appendix G](#) for details on setting the database to single-user mode.
2. Perform a full backup of your SQL Server 2000 database, as we did in Figure 8. See [Appendix G](#) for details.

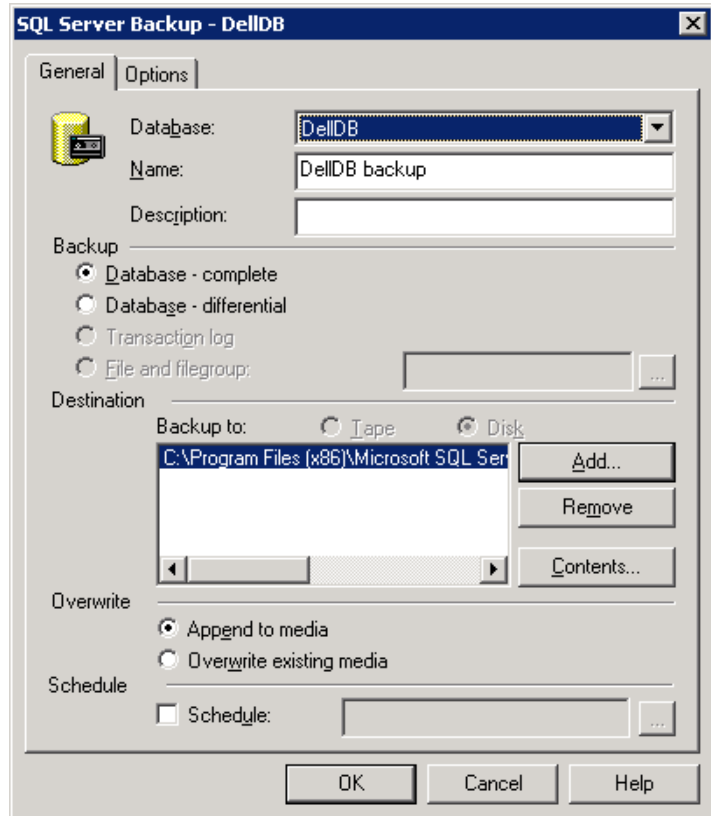


Figure 8. SQL Server 2000 backup configuration.

➔ **NOTE:** If the time a full backup requires is unacceptable given your migration maintenance schedule, you can take the full backup at a previously scheduled time, and at this point of the migration take only a differential or transaction log backup. If the transaction log size is small relative to the data file size, this approach can decrease migration time significantly. If you go this route, be certain to keep your backup chain intact.

### Restoring your database to the SQL Server 2008 R2 virtual machine

On the SQL Server 2008 R2 virtual machine, take the following steps:

1. Open the virtual machine console, and log into the virtual machine.
2. Open SQL Server Management Studio, connect to the SQL Server 2008 R2 instance, and restore the database. (Restore times vary.) See [Appendix G](#) for details.

➔ **NOTE:** At this step, you must give your new database the same name as your SQL Server 2000 database. Changing the name could break applications that refer to the database by name.

3. While in SQL Server Management Studio, reset the database access property to multi-user, and set the compatibility level to SQL 2008 (level 100). See [Appendix G](#) for details.
4. Modify application connection strings or change Domain Name System (DNS) pointers that reference the migrated database.

### **Moving your SQL Server 2005 database: performing the backup**

The process for migrating your SQL Server 2005 databases is almost identical to SQL Server 2000, with some exceptions in the newer management tool, SQL Server Management Studio. Using the SQL Server 2005 server, take the following steps to back up your database. Backup times vary. Again, note that you must do any special planning in advance to minimize your downtime window.

1. To keep users from issuing updates during the migration process, you can either set the database to read-only mode or set the access property to SINGLE\_USER. Both choices immediately sever all user connections. See [Appendix G](#) for details on setting the database to single-user mode.
2. Perform a full backup of your SQL Server 2005 database, as we did in Figure 9. See [Appendix G](#) for details.

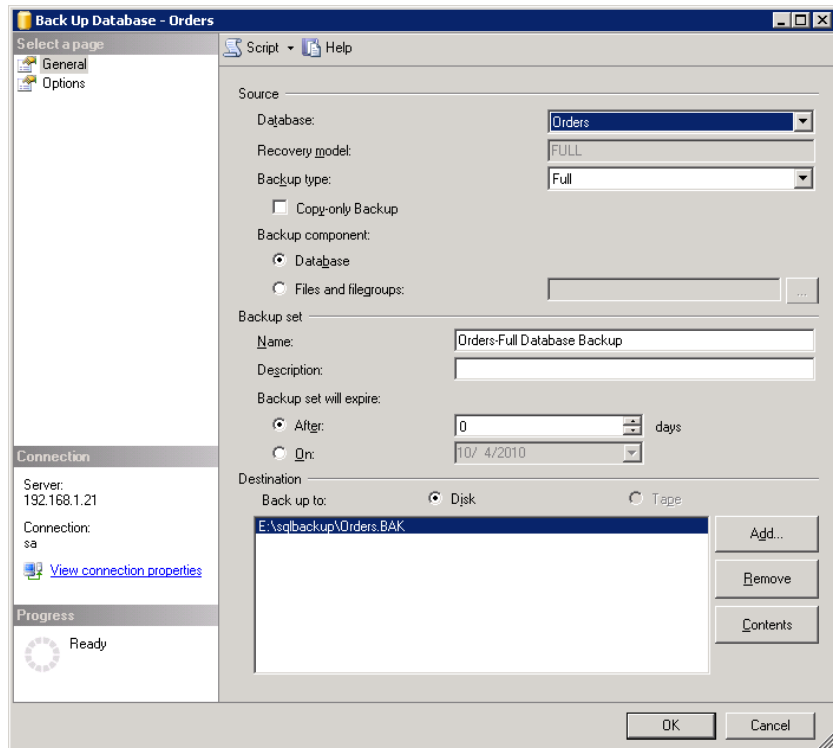


Figure 9. SQL Server 2005 backup configuration.



**NOTE:** If the time a full backup requires is unacceptable given your migration maintenance schedule, you can take the full backup at a previously scheduled time, and at this point of the migration take only a differential or transaction log backup. If the transaction log size is small relative to the data file size, this approach can decrease migration time significantly. If you go this route, be certain to keep your backup chain intact.

### Restoring your database to the SQL Server 2008 R2 virtual machine

On the SQL Server 2008 R2 virtual machine, take the following steps:

1. Open the virtual machine console, and log into the virtual machine.
2. Open SQL Server Management Studio, connect to the SQL Server 2008 R2 instance, and restore the database. (Restore times vary.) See [Appendix G](#) for details.



**NOTE:** At this step, you must give your new database the same name as your SQL Server 2005 database. Changing the name could break applications that refer to the database by name.

3. While in SQL Server Management Studio, reset the database access property to multi-user, and set the compatibility level to SQL 2008 (level 100). See [Appendix G](#) for details.
4. Modify application connection strings or change Domain Name System (DNS) pointers that reference the migrated database.

## WE SHOW YOU HOW: AFTER THE MOVE

After you have completed your side-by-side migration, you typically need to perform some post-migration tasks. Your specific list of tasks will depend heavily on your pre-migration research and planning. In this section, we briefly discuss several of the most common tasks.

### Logins and dependencies

#### Windows and SQL Server logins

As in previous versions of SQL Server, there are two methods of authenticating to SQL Server 2008 R2: Windows logins and SQL Server logins. You create and administer Windows logins at either the operating system level or the Active Directory domain level, and you can assign those logins' rights to SQL Server resources. You create and manage SQL Server logins, however, within SQL Server. The processes for extracting login information and creating the transferred login entities on the migration server is very similar for both Windows-authenticated and SQL Server-authenticated logins. You should, however, take some extra steps to ensure a smooth migration for SQL Server logins.

Below, we describe how to script both Windows-authenticated logins and SQL Server-authenticated logins to a query window and recreate them on the new SQL Server 2008 R2 virtual machine.

Please see [Appendix H](#) for details on transferring Windows logins and [Appendix I](#) for details on transferring SQL Server logins.

To transfer logins, take the following steps on the SQL Server 2008 R2 virtual machine. Note that steps 1 through 6 apply to both Windows logins and SQL Server logins, while Step 7 refers solely to SQL Server logins.

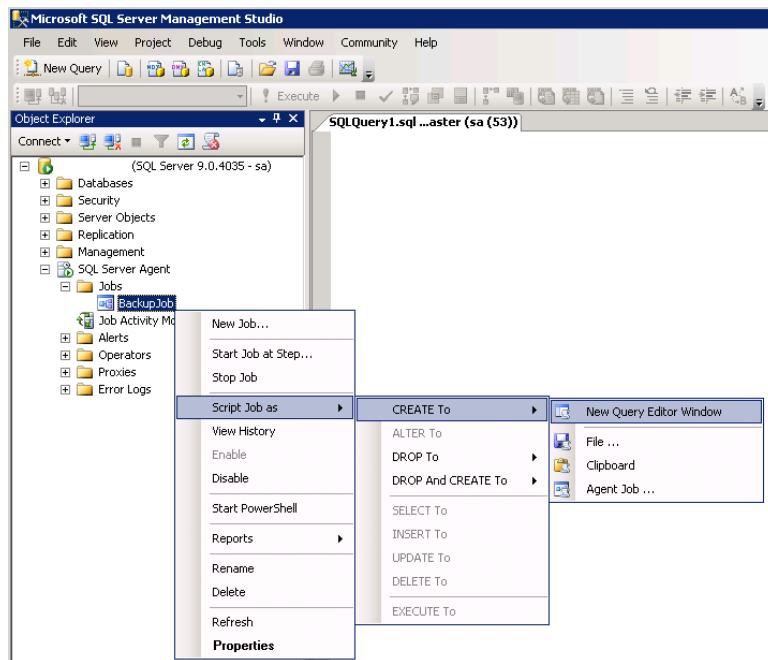
1. Log into the virtual machine.
2. In SQL Server Management Studio, in the Object Explorer pane, connect to **both** your SQL Server 2000 server and your SQL Server 2008 R2 virtual server. Be sure to have the Object Explorer Details tab open (View→Object Explorer Details).
3. Expand the tree view of the SQL Server 2000 server, browse to the Security tab, and click the Logins node. In the Object Explorer Details, you will now see a list of all logins on the SQL Server 2000 server.
4. If necessary, use the sorting and filtering options in the Object Explorer Details tab, and take note of which logins you would like to migrate. Select them by clicking; use the standard Windows controls (Ctrl key, Shift key, etc.) to select multiple logins.
5. Right-click the logins you selected, and choose Script Login As→Create To→New Query Window. Be sure to change the connection of this new query window to connect to your new SQL Server 2008 R2 virtual server by right-clicking and selecting Change Connection.
6. Execute the script on your SQL Server 2008 R2 virtual server to create the logins. If you are transferring Windows-based logins, the process is complete. If you are transferring SQL Server logins, continue with Step 7.
7. For SQL Server logins, the script you executed in Step 6 creates the login, marks it as disabled, and assigns it a random password, but the script does not map the login to the database user. To avoid having a database user that is “orphaned” from a login, use the `sp_change_users_login` stored procedure, to view orphaned users and to map a user to a login. [Appendix I](#) provides an example of this process.

### SQL Server Agent jobs

Almost all installations schedule SQL Server Agent jobs that run against their databases, such as backups, index rebuilds, and other maintenance items. You must migrate these jobs to your new SQL Server 2008 R2 virtual machine. Log into the virtual machine, open SQL Server Management Studio, and connect to your older SQL



Server 2000 server. Locate the jobs you need to migrate in SQL Server Management Studio (on your SQL Server 2000 server) under the SQL Server Agent, right-click them, and choose Script Job As → Create To → New Query Window (see Figure 10). Connect to the SQL Server 2008 R2 virtual server, and run the resulting script in a query window. Verify the job was created successfully in SQL Server Management Studio.



**Figure 10. Scripting out SQL Agent jobs in SQL Server Management Studio.**

### Other external dependencies

Your pre-migration research may well have yielded a list of items for you to implement now; the list might include references to local or remote file shares, database mail or SQL Mail configurations, stored procedures in system databases, and/or linked servers. Addressing these external dependencies will ensure a smooth finish to your migration.

## SUMMING IT ALL UP

The Dell PowerEdge R515, powered by Microsoft's Windows Server 2008 R2 operating system with Hyper-V virtualization technologies and paired with SQL Server 2008 R2, offers a compelling case for consolidation of your SQL Server resources from a legacy SQL Server 2000 and SQL Server 2005 environment to a new environment with SQL Server 2008 R2. As this Guide has explained, the migration process is quite simple, consisting of careful planning, simple backup and restore procedures, and post-migration verifications. In addition, the tools provided by SQL Server 2008 R2 simplify migrating your databases. With careful planning, you can execute migrations with little difficulty and with little to no interruption of service to users.

## APPENDIX A – EXAMPLE DATABASE SURVEY

In this appendix, we provide an example survey of detailed information you may want to gather about the SQL Server instances and databases you are targeting for consolidation. While this survey is a good starting point, it may not contain every relevant detail for your particular environment.

### 1. Hardware level

#### a. CPU

- i. Vendor
- ii. Model
- iii. Number of cores per socket

#### b. Memory

- iv. Total quantity in GB
- v. Memory speed

#### c. Storage

- vi. Quantity, both current and projected growth
- vii. Disk speed
- viii. Repository type (Fibre Channel SAN, iSCSI SAN, SCSI direct-attached)
- ix. Disk interface (SCSI, SAS, etc.)

### 2. OS level

- a. Server name
- b. IP address, subnet, etc.
- c. Domain information
- d. Operating system version, build number, and service pack
- e. Drive letter information and layout

### 3. SQL Server instance level

- a. Whether it's clustered, version, edition (Workgroup, Standard, Enterprise), 64-bit or 32-bit (and if 32-bit, with or without AWE), and service pack level
- b. Authentication mode (Windows only or Mixed Mode)
- c. Instance name (if not a default instance)
- d. SQL port number (e.g., is it the default 1433 or another port? If a named instance, what is the port?)
- e. Communication protocol (named pipes or TCP/IP)

- f. Service account and all service permission information (does SQL Server Agent run under a different service account?)
  - g. Are there any non-default master or model database objects?
  - h. Are there any linked server objects?
  - i. Are other SQL modules involved or dependent on this instance (e.g., Analysis Services, Reporting Services, etc.)?
  - j. Default installation directories, data directories, and log directories
  - k. Tempdb – highly volatile? Slightly volatile? Medium usage?
4. Database level
- a. Database properties and options
    - i. Recovery model, auto-shrink, etc.
    - ii. Files and filegroups (size and location)
  - b. Backups
    - i. How many full backups, and on what schedule?
    - ii. How many differential backups, and on what schedule?
    - iii. How many log backups, and on what schedule?
    - iv. Current backup window?
    - v. Can backup window shift to accommodate the consolidated backup window?
  - c. Other database issues
    - vi. Is this database in a replication configuration?
    - vii. Is this database a partner in a mirroring session (2005/2008 only)?
    - viii. Do any SSIS/DTS packages reference this database?
    - ix. Do any SQL Server Agent jobs reference this database?
  - d. Users/Logins
    - x. Number of users this database affects
    - xi. Would this database require a login whose name conflicts with another name on the target consolidation server? If so, you would need to create the login and map the database user using `sp_change_users_login`

## APPENDIX B – PREPARING THE STORAGE

For our sample migration, we used only internal storage on the Dell PowerEdge R515. On our server, we configured two disks in a RAID 1 configuration and the remaining six disks in a RAID 10 configuration. This created a C: drive and E: drive in our host operating system. We stored all virtual machine files and virtual hard drives on the E: drive, our six disk RAID 10 volume. Below, we cover the specifics on configuring the internal storage.

### Configuring the internal storage

1. During the boot process, Press Ctrl+R to enter the PERC BIOS Configuration Utility.
2. Highlight the controller, press F2, and choose to create the new virtual disk.
3. Provide the desired RAID setting, such as RAID 1, 5, 10, and so on. Choose the relevant disks for this volume.
4. Press OK to complete the volume configuration.
5. Highlight the new virtual disk, press F2, then Initialization, and then Start Init. This will fully initialize your virtual disk.
6. Repeat steps 2 through 5 for the remaining virtual disks in your server.

## APPENDIX C – INSTALLING WINDOWS AND ENABLING HYPER-V

We installed Windows Server 2008 R2 Standard Edition on the Dell PowerEdge R515 with all default options. You may wish to perform additional preparation steps as part of your specific environment, such as standardizing the machine name, enabling certain firewall ports, and so on.

### Installing the operating system

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 Standard (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 to create a new partition.
9. Click Apply to apply the changes.
10. Select the newly created partition, and click Format. After the format completes, click Next.
11. At the User's password must be changed before logging on warning screen, click OK.
12. Enter the desired password for the administrator 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 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.

## Enabling the Hyper-V role

Follow the steps below to enable Hyper-V on Windows Server

2008 R2.

1. Click Start→Administrative Tools→Server Manager.
2. Right-click Roles, and select Add Roles.
3. On the Before You Begin page, click Next.
4. On the Select Server Roles page, select Hyper-V, and click Next.
5. On the Hyper-V page, click Next.
6. On the Create Virtual Networks page, select the appropriate Local Area Connection, and click Next. In our Dell PowerEdge R515, we had two onboard NICs and four add-in NICs in the system. We used one onboard port as a management port, and two add-in NICs for the virtual machine traffic.
7. On the Confirm Installation Selections page, click Install.
8. On the Installation results screen, click Close.
9. When Windows prompts you to reboot the server, click Yes. The server will reboot twice.
10. On the Installation Results page, click Close.

## APPENDIX D – CREATING VIRTUAL MACHINES

Below, we provide detailed steps on creating your virtual machines using Hyper-V manager. Note that drive availability and required number of VHD files may vary based on your environment.

1. Click Start→Administrative Tools→Hyper-V Manager.
2. From the Action menu, select New→Virtual Hard Disk.
3. Choose Fixed Size, assign 25GB for size, and place the VHD in the appropriate folder on the server. We chose to store all of our VHD files on our six disk RAID 10 volume.
4. Repeat Step 3 for the remaining VHDs necessary for your VMs. On our SQL Server VMs, for flexibility, we created a VHD for the OS, a VHD for SQL Server data, a VHD for SQL Server logs, and a VHD for backup and utility files.
5. From the Action menu, select New→Virtual Machine.
6. On the Before You Begin page, click Next.
7. On the Specify Name and Location page, enter the name for your new virtual machine, and click Next.
8. On the Assign Memory page, enter the RAM amount.
9. On the Configure Networking, choose the network assigned to VM traffic, and click Next.
10. Choose to add a virtual hard disk later.
11. On the Installation Options page, accept the default of Install an operating system later, and click Next.
12. On the Completing the New Virtual Machine Wizard page, click Finish.
13. Right-click the virtual machine, and choose Settings.
14. Click Processors, and choose the number of virtual processors. We chose two virtual processors for our database virtual machines.
15. Click the virtual IDE controller 0.
16. Click Add, and click Hard Drive.
17. Browse to the first VHD you created in Step 4, and choose that VHD for the guest OS VHD.
18. Click Apply.
19. Click the SCSI controller.
20. Click Add, and click Hard Drive.
21. Browse to the second VHD you created in Step 4, and choose that VHD for the SQL Server data VHD.
22. Click Apply.
23. Repeat steps 19 through 22 for the remaining SQL Server related VHDs.



24. Power on the VM, and then proceed to install Windows Server 2008 R2 on each VM, following the instructions from the above “Installing the operating system” section.

## APPENDIX E – INSTALLING SQL SERVER 2008 R2

Install an instance of Microsoft SQL Server 2008 R2 by following these steps. This installation walkthrough covers only the installation of the Database Engine and Management Components. For other components outside the scope of this guide, such as Reporting Services, Integration Services, or Analysis Services, see Microsoft documentation at [http://msdn.microsoft.com/en-us/library/ms143219\(SQL.105\).aspx](http://msdn.microsoft.com/en-us/library/ms143219(SQL.105).aspx).

1. On the Dell PowerEdge R515, click the Start button, and click Hyper-V Manager.
2. Right-click the virtual machine, and choose to connect.
3. Log into the virtual machine.
4. Insert the installation DVD for SQL Server 2008 R2 into the physical server's DVD drive.
5. Using the Hyper-V menu, choose Media, and Capture D: Drive, where D: is the letter of the physical server's DVD drive.
6. If Autoplay does not begin the installation, navigate to the SQL Server 2008 R2 DVD, and double-click.
7. If the installer prompts you with a .NET installation prompt, click Yes to enable the .NET Framework Core role.
8. At the SQL Server Installation Center screen, click Installation. Click New installation or add features to an existing installation.
9. At the Setup Support Rules screen, click OK.
10. At the Product Key screen, enter your licensing information, if applicable, and click Next.
11. At the License Terms screen, accept the license terms, and click Next.
12. At the Setup Support Files screen, click Install.
13. At the Setup Support Rules screen, click Next.
14. At the Setup Role screen, choose SQL Server Feature Installation, and click Next.
15. At the SQL Server 2008 R2 Feature Selection screen, select the features that your organization requires. We chose the following features for this guide: Database Engine Services, Client Tools Connectivity, Client Tools Backwards Compatibility, Management Tools – Basic, and Management Tools – Complete. Click Next.
16. At the Installation Rules screen, click Next.
17. At the Instance Configuration, enter the appropriate details for your configuration. For a default instance, leave the defaults

selected. For a named instance, enter a new instance name and adjust the file paths as necessary.

- 18.** At the Disk Space Requirements screen, click Next.
- 19.** At the Server Configuration screen, choose the service account, preferably an Active Directory domain account, fill in a password if necessary, and click Next.
- 20.** At the Database Engine Configuration screen, choose an authentication mode. If your legacy servers use SQL Server logins at all, select Mixed Mode. If you exclusively use Active Directory domain accounts in your SQL Server environment, choose Windows Authentication.
- 21.** If you choose to use Mixed Mode authentication, enter a password for the system administrator (SA) account, click Add Current User, and click Next.
- 22.** At the Error Reporting screen, click Next.
- 23.** At the Installation Configuration Rules screen, click Next.
- 24.** At the Installation screen, click Install.
- 25.** At the Complete screen, click Close.
- 26.** After the SQL Server 2008 R2 installation process completes, check Microsoft's Web site for the latest SQL Server service pack and install that service pack. At the time we wrote this Guide, there were none.

## APPENDIX F – INSTALLING UPGRADE ADVISOR

In this appendix, we walk through the steps to install and run the SQL Server 2008 R2 Upgrade Advisor and save Upgrade Advisor reports.

1. Insert the SQL Server 2008 R2 DVD to the machine in which you wish to install the Upgrade Advisor. On the Planning screen, click Install SQL Server Upgrade Advisor.
2. Click Next to begin the installation wizard, accept the licensing terms, and click Next.
3. Click Next to accept the default Registration information, click Next to accept the default installation path, and click Next to begin the installation.
4. At the Completing the Microsoft SQL Server 2008 R2 Upgrade Advisor screen, click Finish to exit the setup.
5. Start the Upgrade Advisor by selecting Start→All Programs→Microsoft SQL Server 2008 R2→SQL Server 2008 R2 Upgrade Advisor.
6. Click the Launch Upgrade Advisor Analysis Wizard link.
7. At the Welcome screen, click Next.
8. On the SQL Server Components screen, by default the Upgrade Advisor populates the server name field with the local computer name. If you need to scan a remote server, type the computer name or IP address of the SQL Server you want to analyze and do one of the following:
  - a. Click Detect to allow the Upgrade Advisor to scan the components on the SQL Server instance you specified. (We chose this option.)
  - b. Select the components of the SQL Server instance you would like the Upgrade Advisor to scan.
9. On the Connection Parameters screen, specify the Authentication Type and credentials if necessary.
10. On the SQL Server parameters screen, select one or more databases and, if you so desire, supply a trace file or SQL script file to analyze. Should you wish to use a trace file, you must generate it before you begin the Upgrade Advisor. For instructions about how to generate a trace file, see [http://msdn2.microsoft.com/en-us/library/ms187929\(SQL.105\).aspx](http://msdn2.microsoft.com/en-us/library/ms187929(SQL.105).aspx).
11. If you specified analysis of Data Transformation Packages (or if the detection process discovered DTS on the SQL Server 2000 server), you should now choose whether you want the Upgrade Advisor to scan for (a) DTS packages on the SQL Server 2000 server or (b) DTS package files stored on the file system. Click Next to continue.

12. On the Confirm Upgrade Advisor Settings screen, review your choices, and click Run.
13. When the analysis completes, click Launch Report to view the analysis report.
14. You can now use the drop-down filter tools to view report items by component or by issue severity.



**NOTE:** If you want to export the report, click Export Report in the lower right corner of the Upgrade Advisor interface.

## APPENDIX G – MIGRATING DATABASES

In this appendix, we give detailed instructions about a basic side-by-side migration of a user database from a SQL Server 2000 or SQL Server 2005 server to a SQL Server 2008 R2 virtual server. There are multiple methods of accomplishing this task, including using TSQL commands and automated tools. Here, we discuss performing the database migration using the graphical interface tools Microsoft provides with SQL Server 2000, SQL Server 2005, and SQL Server 2008 R2.

### Backing up a SQL Server 2000 database

1. Log into the SQL Server 2000 server as either an administrative user or a user with full rights on SQL Server 2000.
2. Start Enterprise Manager by selecting Start→All Programs→Microsoft SQL Server→Enterprise Manager.
3. Expand Microsoft SQL Servers→SQL Server Group→(local) (Windows NT)→Databases.
4. Set the database to single-user mode to ensure no updates occur. Right-click the database, and choose properties. Click the Options tab, check the Restrict Access checkbox, and choose Single User.



**NOTE:** Setting the database to single-user mode immediately disconnects all users. You must notify your users well in advance of setting the database to single-user mode.

5. Right-click the database, and select All Tasks→Backup Database.
6. Keep the default of complete backup.
7. Click Add... to add a backup device, and click the “...” button to browse to the backup location you want to use. Select a folder, and enter a filename, such as DatabaseNameMigrationBackup.bak.
8. Click OK to close the Backup Device Location window, and click OK to close the Select Backup Destination window.



**NOTE:** To simplify moving the database later, we recommend creating the backup in a shared local folder.

9. Optionally, on the Options tab, select the checkbox to verify your database backup upon completion.



**NOTE:** This option increases backup time but checks the integrity of the backup file. We chose to verify the backup.

10. Click OK to begin the SQL Server 2000 database backup. (Backup times vary with database size and backup device configuration.)
11. After the backup completes, in Enterprise Manager, under Databases, right-click the database, and choose All Tasks→Take Offline. The old database is now offline, so SQL Server 2000 will not allow any connections to it.
12. Log out of Windows on the SQL Server 2000 server.

## Backing up a SQL Server 2005 database

1. Log into Windows on the SQL Server 2005 server as either an administrative user or a user with full rights on SQL Server 2005.
2. Start SQL Server Management Studio by selecting Start→All Programs→Microsoft SQL Server 2005→SQL Server Management Studio.
3. Enter your server name, and select Windows Authentication.
4. Set the database to single-user mode to ensure no updates occur.



**NOTE:** Setting the database to single-user mode immediately disconnects all users, causing them to lose any work in progress. You must notify your users well in advance of setting the database to single-user mode.

5. To set the database to single-user mode, in Object Explorer, right-click the database, and choose properties.
6. Click Options, and scroll to the State section.
7. For the Restrict Access setting, choose SINGLE\_USER, and click OK. Click Yes when prompted that other connections will be cut off.
8. Right-click the database, and choose Tasks, Back Up.
9. Click Add... to add a backup device, and click the "..." button to browse to the backup location you want to use. Select a folder, and enter a filename, such as DatabaseNameMigrationBackup.bak.
10. Click OK to close the Backup Device Location window, and click OK to close the Select Backup Destination window.



**NOTE:** To simplify moving the database later, we recommend creating the backup in a shared local folder.

11. Optionally, on the Options tab, select the checkbox to verify your database backup upon completion.



**NOTE:** This option increases backup time but checks the integrity of the backup file. We chose to verify the backup.

12. Click OK to begin the SQL Server 2005 database backup. (Backup times vary with database size and backup device configuration.)
13. After the backup completes, in SQL Server Management Studio, under Databases, right-click the database, and choose All Tasks→Take Offline. The old database is now offline, so SQL Server 2005 will not allow any connections to it.
14. Log out of Windows on the SQL Server 2005 server.

## Restoring a database to SQL Server 2008 R2

1. Transfer either your SQL Server 2000 or SQL Server 2005 backup file to the virtual machine.
2. Log into Windows on the virtual machine running SQL Server 2008 R2.
3. In Windows Explorer, navigate to backup folder location.
4. Open SQL Server Management Studio by selecting Start→All Programs→Microsoft SQL Server 2008 R2→SQL Server Management Studio.
5. Select Database Engine for Server type, and type the server name in Server name. For Authentication, we used Windows Authentication.
6. In the Object Explorer pane, right-click Databases, and select Restore Database.
7. Enter the name of the database in the To database box.



**NOTE:** It is critical to use the same database name as on your SQL Server 2000 or SQL Server 2005 server. If the database name does not match, applications that depend on the database name may break.

8. Click the From device button, and click the “...” button to browse to the local folder where you copied the backup file.
9. To add the backup file location, click Add. Browse to the appropriate folder, find the backup file, select it, and click OK.
10. To return to the Restore Database window, click OK.



11. Select the checkbox that now appears in the Restore column. Optionally, click Options on the left, and ensure the file locations are appropriate in the Restore As column.



**NOTE:** In your new virtual machine, your storage layout may differ significantly from your older physical machine. Modify the file layout on the Options screen if necessary.

12. To begin the restore, click OK. Note the progress indicator in the lower left of the Restore window. (Restore time varies with database size and server and with disk subsystem speed.)
13. After the restore is complete, you must change the database access state back to multi-user and upgrade the compatibility level by performing the following steps:
  - a. Right-click the database in Object Explorer, and select Properties.
  - b. On the left side to access database options, click Options.
  - c. Change Compatibility level to SQL Server 2008 (100).
  - d. Scroll down to Restrict Access, and change to MULTI\_USER.
  - e. Click OK.
  - f. Click Yes to agree to shutting down other connections.

## APPENDIX H – TRANSFERRING WINDOWS LOGINS

In this appendix, we provide detailed instructions for transferring Windows-based logins from a SQL Server 2000 or SQL Server 2005 machine to SQL Server 2008 R2 in an Active Directory or Windows login-only environment. For information about transferring logins when using SQL Server authentication, see [Appendix I](#).

1. On the virtual machine running SQL Server 2008 R2, log into Windows, then open SQL Server Management Studio by selecting Start→All Programs→Microsoft SQL Server 2008 R2→SQL Server Management Studio.
2. On the Connect to Server screen, select Database Engine for the server type, enter the computer name of the relevant older server, either your SQL Server 2000 machine or your SQL Server 2005 machine, and click Connect.
3. Select View→Object Explorer Details to bring up the Object Explorer Details tab.
4. In the Object Explorer pane, expand the Security folder, and highlight the Logins folder.
5. Add the Login Type column to the display by right-clicking the column-header area in the Object Explorer Details pane and selecting Login Type. Now Windows Logins appear as Login Type 0. Optionally, use the column headers to sort by Default Database, or use the filter button to find only logins that you are interested in migrating.
6. Using the standard Windows multi-select key combinations (Ctrl key or Shift key), highlight the logins you are interested in migrating.
7. Right-click the highlighted logins, and select Script Login as→Create To→New Query Editor.
8. On the Connect to Database Engine screen, change the Server name to your SQL Server 2008 R2 virtual machine, and click Connect.
9. Click Execute to run the resulting script.

## APPENDIX I – TRANSFERRING SQL SERVER LOGINS

In this appendix, we give detailed instructions on transferring SQL Server logins from a SQL Server 2000 or SQL Server 2005 machine to a virtual machine running SQL Server 2008 R2. For information about transferring logins when using Windows authentication, see [Appendix H](#).

1. On the virtual machine running SQL Server 2008 R2, log into Windows, and open SQL Server Management Studio by selecting Start→All Programs→Microsoft SQL Server 2008 R2→SQL Server Management Studio.
2. On the Connect to Server screen, select Database Engine for the server type, enter the computer name of the relevant older server, either your SQL Server 2000 machine or your SQL Server 2005 machine, and click Connect.
3. Select View→Object Explorer Details to bring up the Object Explorer Details tab.
4. In the Object Explorer pane, expand the Security folder, and highlight the Logins folder.
5. Add the Login Type column to the display by right-clicking the column-header area and selecting Login Type. Now SQL Server Logins appear as Login Type 2. Optionally, use the column headers to sort by Default Database, or the Filter button to find only logins that you are interested in migrating.
6. Using the standard Windows multi-select key combinations (Ctrl key or Shift key), highlight the logins you are interested in migrating.
7. Right-click the highlighted logins, and select Script Login as→Create To→New Query Editor Window.
8. On the Connect to Database Engine screen, change the Server name to your SQL Server 2008 R2 virtual server, and click Connect.
9. Click Execute to run the resulting script.

### Mapping transferred logins to database users

After transferring SQL Server logins from server to server, you must map those logins to the database users that you migrated during the restore process.



**NOTE:** This process assumes that you have already migrated the applicable database to the new server.

1. On the SQL Server 2008 R2 virtual server, log into Windows. Then open SQL Server Management Studio by selecting Start→All Programs→Microsoft SQL Server 2008 R2→SQL Server Management Studio.
2. On the Connect to Server screen, select Database Engine for the server type, enter the computer name of your SQL Server 2008 R2 virtual server, and click Connect.
3. Right-click the relevant database, and select New Query.
4. Enter the following command, and click Execute:

```
EXEC sp_change_users_login 'Report';
```

5. For each user output from the above step, map these orphaned users to the appropriate matching SQL Server login by taking the following steps:
  - a. Open a query window on the SQL Server 2008 R2 virtual server using steps 1 through 3 above.
  - b. For each login you wish to map, enter the following command, and click Execute:

```
EXEC sp_change_users_login  
'Auto_Fix', 'SQLLogin';
```

- c. The above command maps the database user “SQLLogin” to the SQL Server login of “SQLLogin”. This procedure assumes that the database user and SQL Server login have the same value.
6. To reset the password and enable the account, enter the following command, and click Execute:

```
USE [master]  
  
GO
```

```
ALTER LOGIN [SQLLogin] WITH  
PASSWORD=N'Password1'  
  
GO
```

```
ALTER LOGIN [SQLLogin] ENABLE  
  
GO
```

## 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.



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

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

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

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

---

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

---

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

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

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

---