

CONSOLIDATION



**CONSOLIDATING SQL SERVER® 2000
ONTO DELL™ POWEREDGE™ R900 AND
POWEREDGE™ R905 USING
MICROSOFT'S HYPER-V™**

A Principled Technologies report commissioned by Dell

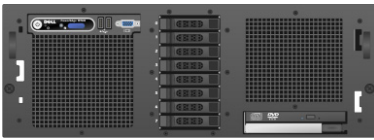


Table of contents

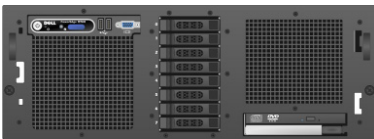
Table of contents	2
Introduction	3
Overview of this companion Guide.....	3
Continuing the migration and consolidation process	4
Creating and configuring your virtual machines	5
Creating a virtual machine	5
Installing Windows Server 2003 in your virtual machine	7
Installing SQL Server 2000 in your virtual machine	9
Moving to your virtual environment	12
Moving your database	12
Transferring logins	13
Summing up	15
Appendix A. Creating the virtual machine and installing the guest OS	16
Creating the virtual machine	16
Adding processors and virtual hard drives to the virtual machine	16
Installing the Windows Server 2003 SP2 guest OS.....	17
Setting IP address, subnet mask, and domain information on your VM	19
Configuring virtual data drives	20
Appendix B. Installing SQL Server 2000 Enterprise Edition.	21
Installing SQL Server 2000 with SP4.....	21
Moving tempdb files from the default location	23
Appendix C. Migrating SQL Server 2000 databases from your physical machine to your virtual machine	25
Appendix D. Transferring logins	27
About Principled Technologies	29

Introduction

Overview of this companion Guide



Dell PowerEdge R900



Dell PowerEdge R905

This Guide is a companion to *How to Guide: Consolidating multiple SQL Server[®] systems onto Dell[™] PowerEdge[™] Servers using Microsoft's Hyper-V[™]*. That Guide reviews general virtualization and consolidation concepts and demonstrates the setup and basic configuration of the Dell[™] PowerEdge[™] R900 or PowerEdge[™] R905 server. Building on that base, this Guide adds the techniques you need to successfully consolidate your Microsoft[®] SQL Server[®] 2000 databases from multiple machines onto a single Windows Server[®] 2008 server system using Microsoft's Hyper-V[™]. In this Guide, we use the Dell PowerEdge R900 and PowerEdge R905 as target servers.

Principled Technologies[®] (PT) has performed hands-on testing and research and drawn on real-world experiences to document best practices and help you take advantage of the features of Hyper-V to consolidate multiple SQL Server 2000 instances onto a single physical system.

This document covers the concepts and procedures specific to virtualizing SQL Server 2000 on the Hyper-V platform. It reviews in detail the approach that PT and Dell recommend for a tested and validated configuration of Hyper-V and SQL Server 2000. For general concepts and procedures on setting up your virtualization server with Windows Server 2008 and Hyper-V, see our accompanying Guide, *How to Guide: Consolidating multiple SQL Server[®] systems onto Dell[™] PowerEdge[™] Servers using Microsoft's Hyper-V[™]*. Using this SQL Server 2000 Guide in conjunction with the general Hyper-V overview Guide, you can easily virtualize your SQL Server 2000 environment.

Microsoft does not support SQL Server 2000 running virtualized in Hyper-V. Although we had no problems implementing a SQL Server 2000 installation under Hyper-V, we cannot guarantee it will work in all scenarios, especially if your installation is complex or involves clustering. In all cases, consider your situation and the possible implications of Microsoft's lack of support of this setup carefully before virtualizing SQL Server 2000.

The Dell PowerEdge R900 supports up to 24 logical processors, or cores, and the Dell PowerEdge R905 supports 16 logical processors. The release version of Microsoft Hyper-V supports a maximum of 16 logical processors. See

<http://technet.microsoft.com/en-us/library/cc816844.aspx> for more details on Hyper-V hardware considerations.



NOTE: Microsoft released an update in September 2008 to increase this maximum to 24 logical processors. For more details, and to download this update, visit <http://support.microsoft.com/kb/956710>.

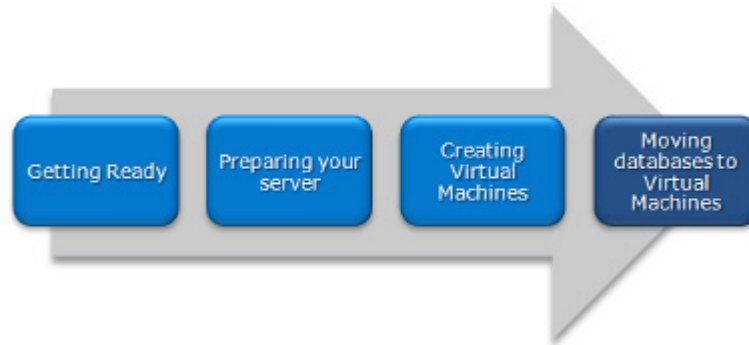
Because we use both the Dell PowerEdge R900 and R905 in this Guide, we refer to the target server as the PowerEdge, and note specifics to each platform where they apply.

Continuing the migration and consolidation process

In the accompanying Guide, *How to Guide: Consolidating multiple SQL Server® systems onto Dell™ PowerEdge™ Servers using Microsoft's Hyper-V™*, we discussed best practices and gave detailed steps on setting up and configuring PowerEdge server with Windows Server 2008 and Hyper-V. We also covered best practices in disk layout as it relates to a SQL Server workload.

We now discuss procedures involved in migrating SQL Server 2000 databases to a virtualized environment. The steps we list below are a continuation of the process we began in the companion Guide. This Guide assumes you have followed the procedures in the previous Guide; the instructions here may not work if you have not followed those steps exactly.

Creating and configuring your virtual machines



Creating a virtual machine

After you have installed the Hyper-V role, the hypervisor will be running and ready for you to create virtual machines. In this section, we provide basic steps for creating new virtual machines, and show how to allocate resources for those machines. For detailed steps, see [Appendix A](#).

1. Open Hyper-V manager under Start | Administrative tools. Connect to your server. Then, choose Action | New | Virtual Machine to start the Virtual Machine creation wizard.
2. Give the virtual machine a name, and provide memory, NIC, and VHD details. Choose the appropriate disk drive to store your new VHD (see Figure 1). In our case, we created a new VHD file on the C: drive of the parent partition.

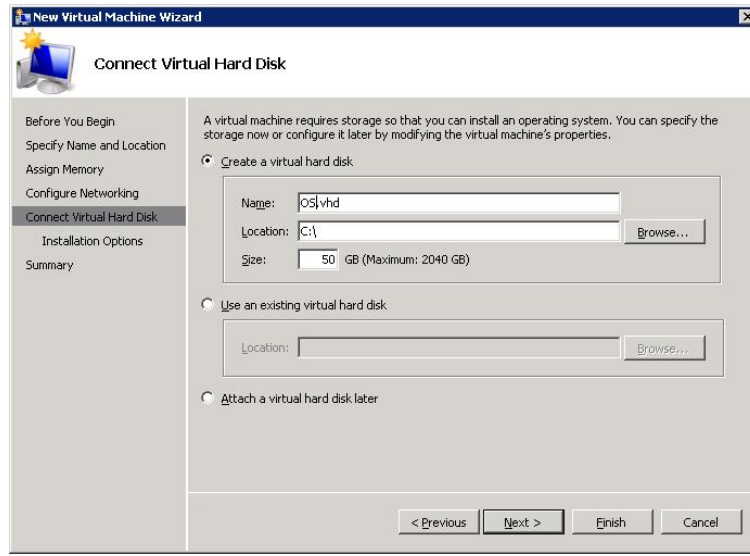


Figure 1. Configuring a Hyper-V virtual machine.

3. Choose Install an operating system later. We cover these steps below in the next section.
4. Finish the wizard, then right-click the virtual machine, choose Settings, and configure multiple processors and other devices as necessary.
5. Finally, create a SCSI controller for this VM. Create additional VHDs, and attach them to this SCSI controller: one for tempdb, one for SQL Server logs, and one for SQL Server user data.

Installing Windows Server 2003 in your virtual machine

To make the virtual machine operable, you will of course need to install an operating system, just as you would on a physical machine. Virtualizing your machine allows you to carry over the same operating system as your physical machine had. In this section, we show how to install Windows Server 2003. For details on this procedure, see [Appendix A](#).



NOTE: Microsoft offers a product called System Center Virtual Machine Manager (SCVMM) that allows you to perform a physical to virtual (P2V) machine migration with just a few clicks. If you have highly customized your configuration and a new install of the operating system and SQL Server would not be appropriate, consider using SCVMM or another third-party product to perform the P2V conversion.

This section provides an overview of the Windows Server 2003 installation process on a Hyper-V virtual machine. For the most part, this installation on the virtual machine is identical to a typical Windows Server 2003 installation on the physical machine. We include approximate wait times for each step. ([Appendix A](#) provides complete, detailed installation instructions.)



NOTE: Plan on at least 70 minutes for installing Windows Server 2003 Enterprise Edition on a Hyper-V virtual machine. Each step below takes at least 1 minute. We provide the approximate amount of time each step takes in parentheses. These times exclude data entry time.

- 1.** Insert the Microsoft Windows Server 2003 Enterprise Edition DVD into the physical DVD drive of your PowerEdge server.
- 2.** Start Hyper-V manager, and connect to the relevant VM. Attach the physical optical device to the virtual machine using the Media menu.
- 3.** Start the VM, and accept the Windows Server license agreement. Create your partition if necessary, format your drive using NTFS, and proceed with the Windows Server 2003 installation steps on your virtual machine (see Figure 2). (12 minutes on the PowerEdge R900; 15 minutes on the PowerEdge R905)

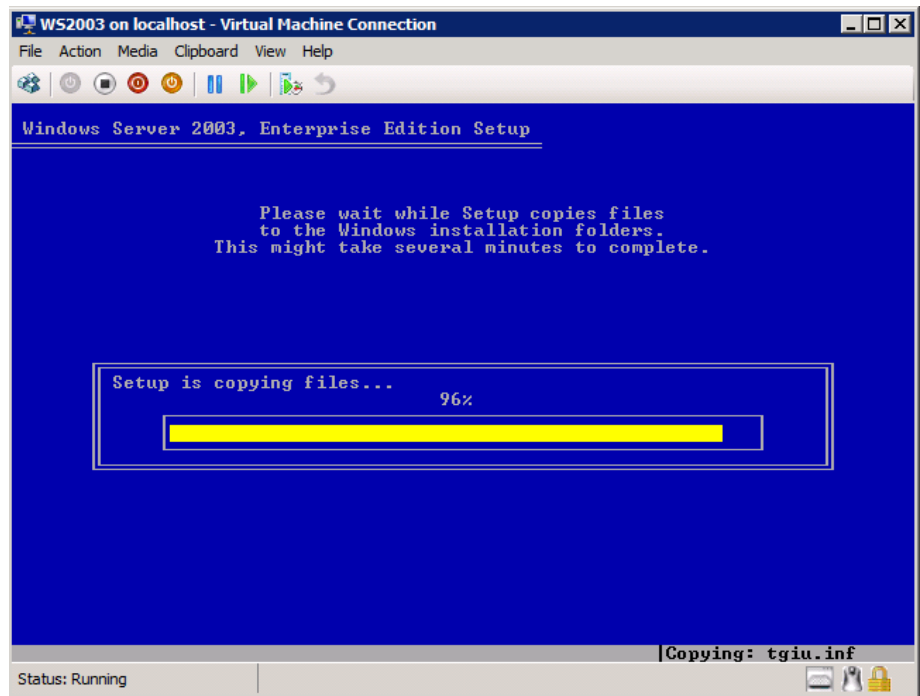


Figure 2. Installing Windows Server 2003 on a virtual machine.

4. Enter your product key and licensing information, set the Administrator password, then set your time and location. Enter any other information the installation software requires. (14 minutes on the PowerEdge R900; 21 minutes on the PowerEdge R905)
5. After the system restarts, complete the installation when the installer prompts you for the second disk. Then, install Windows updates.
6. Enter configuration information, including the IP addresses and domain information, and reboot the system. (3 minutes)
7. Install Service Pack 2. Note: SP2 is a requirement for using the Hyper-V Integration Services. We had previously downloaded Service Pack 2. (8 minutes on the PowerEdge R900; 16 minutes on the PowerEdge R905)
8. Start the Hyper-V Integration Services installation on the virtual machine by choosing Action on the VM menu | Insert Integration Services Setup Disk. Proceed through the Integration Services installation, and reboot the VM if the Integration Services installation requires it. (5 minutes)
9. Configure your disks inside the VM operating system. Follow the same procedures you used when

configuring your physical disks, except that these are virtual disks, connected to the VHD files you created earlier. (3 minutes per disk, though times vary a bit)

Installing SQL Server 2000 in your virtual machine

To migrate your databases from physical machines to virtual machines, you must have installed and properly configured the appropriate version of SQL Server on them. We cover a new installation of SQL Server 2000 in this section. We include approximate wait times for each step. [Appendix B](#) provides detailed installation instructions.

Installing SQL Server 2000 on a Windows Server 2003 virtual machine



NOTE: Plan on at least 15 minutes for installing SQL Server 2000 Enterprise Edition and SQL Server 2000 SP4 on the VM. Each step below takes at least 1 minute. We provide the approximate amount of time each step takes in parentheses. These times exclude data entry time.

- 1.** Insert the SQL Server 2000 Enterprise Edition CD into the physical DVD drive.
- 2.** Start Hyper-V manager, and connect to the relevant VM. Attach the physical optical device to the virtual machine using the Media menu.
- 3.** Open Windows Explorer, and browse to the path on the CD drive where the setup.bat file exists. On our MSDN distribution, this was at
D:\ENGLISH\SQL2000\ENT\
(4 minutes)
- 4.** Double-click setup.bat, and begin the installation. Configure the instance, security, data paths, and then accept the license agreements. (4 minutes)
- 5.** Complete the installation, and then reboot the machine to clear any pending reboot needs (see Figure 3). (2 minutes)

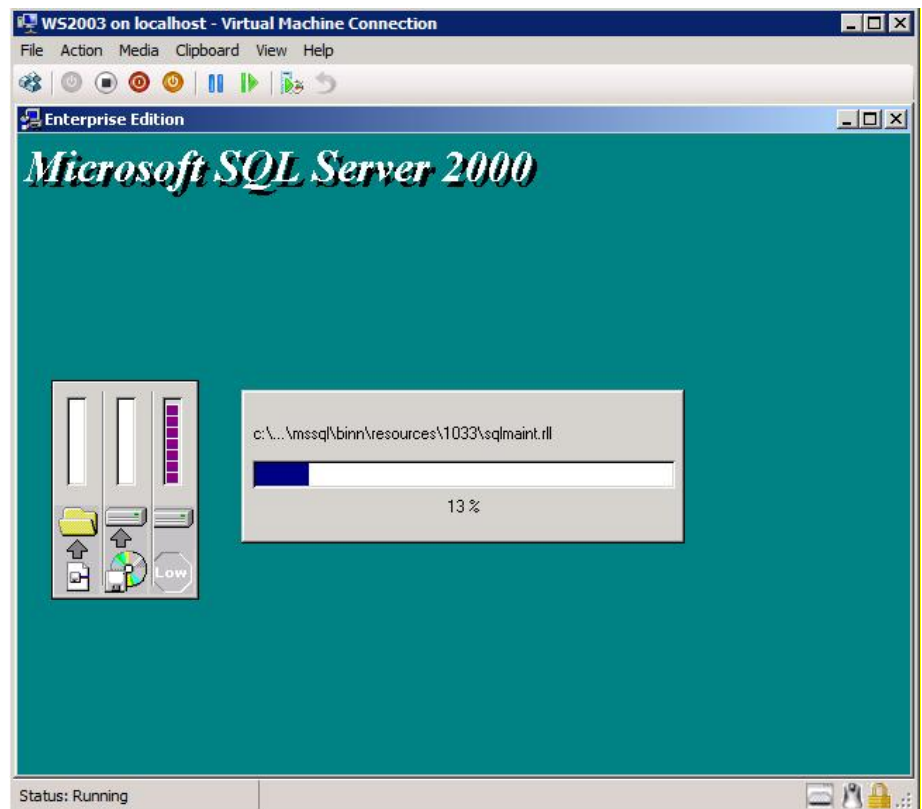


Figure 3. Installing SQL Server 2000 on a virtual machine.

6. Extract the SQL Server 2000 SP4 files to a local directory, browse to that directory, and once again double-click setup.bat to begin the SP4 installation.
7. Accept the defaults on the SP4 screens, and complete the SP4 install. (2 minutes)
8. Reboot the VM once more to apply all service pack files. (2 minutes)



BEST PRACTICE: After you complete the SQL Server installation on the VM, give the SQL Server service account the right to prevent the operating system from paging its memory to disk. SQL Server dynamically allocates and deallocates memory to relieve memory pressure and swapping. However, another process can always request a substantial amount of memory and thus cause the OS to swap some of SQL Server's memory to disk before SQL Server has the chance to react. This setting prevents that.

Windows Server 2003 Enterprise Edition provides a setting that allows SQL Server to retain its data in physical memory instead of

swapping the data to virtual memory, the page file, or disk. This is especially important on 32-bit systems using AWE. To enable this setting, give the account running the SQL Server service the right to lock pages in memory. To do so, follow these steps inside the VM:

1. Click Start.
2. Go to Administrative tools | Local Security Policy.
3. Click Local Policies, and then click User Rights Assignment.
4. Scroll down to Lock pages in memory, and double-click (see Figure 4).
5. Click Add User or Group, and enter the SQL Server service account name.

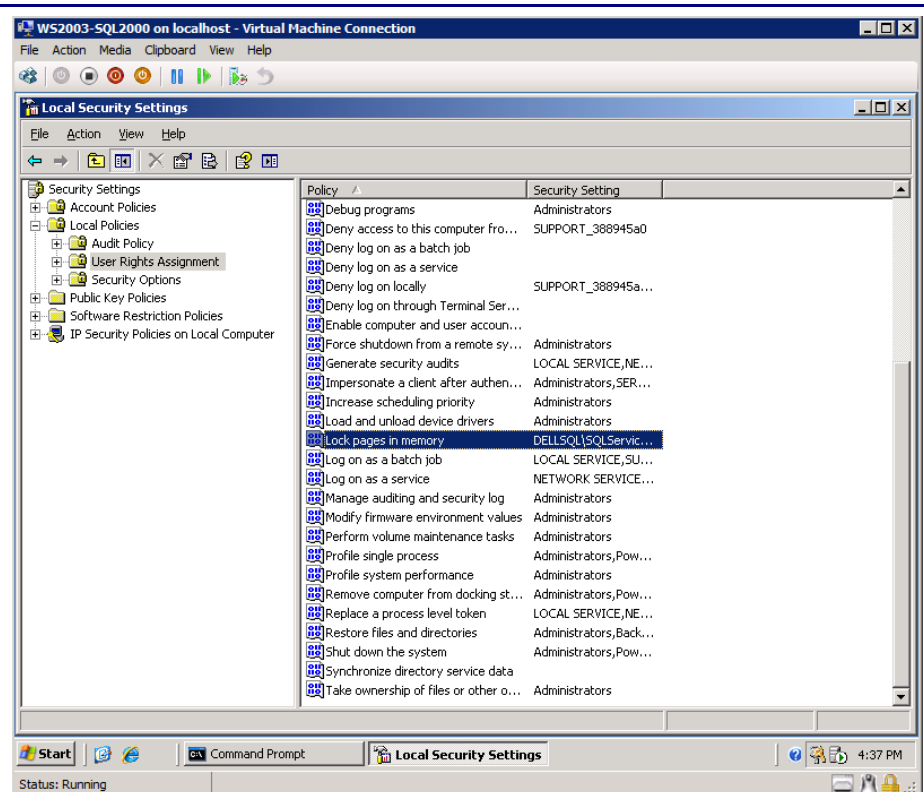
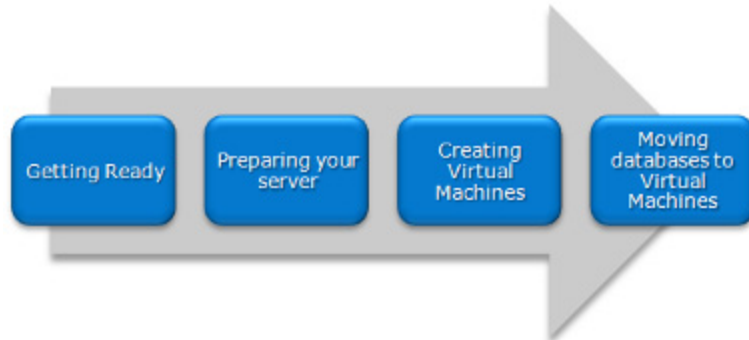


Figure 4. Selecting the Lock pages in memory setting inside your VM.



BEST PRACTICES: Leave the memory settings for SQL Server 2000 at their default, which allows SQL Server 2000 to dynamically configure the memory as needed. Note that if you choose to enable AWE for SQL Server 2000 on the VM, the dynamic management of memory is disabled.

Moving to your virtual environment



Moving your database

In this section, we provide an overview of the processes involved in moving your database from a physical server running SQL Server 2000 to a virtual server running the same version. Our assumption for this Guide is that this move entails no upgrades. In other words, because you have virtualized your servers, you will only be moving SQL Server 2000 databases to your virtual machine running SQL Server 2000. Upgrading from an earlier version of SQL Server is outside the scope of this Guide. We provide detailed instructions in [Appendix C](#).

We performed all SQL Server 2000 backup and restore maintenance using Query Analyzer, the main query tool for SQL Server 2000.



NOTE: You can move databases between instances and versions of SQL Server by various methods, including detach/attach and backup/restore. We chose to use the backup/restore method.

Backing up your SQL Server database on your physical machine

On the original physical server running SQL Server, take the following steps to back up your database. Backup times vary depending on the size of the database.

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 C](#) for details on

setting the database to single-user mode. We chose to set the database to SINGLE_USER mode.

2. Using TSQL and Query Analyzer, perform a full backup of your SQL Server database. See [Appendix C](#) for details.



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

Restoring database to the SQL Server instance on your virtual machine

To restore your database to the VM running SQL Server, perform the following steps:

1. Connect to your VM in Hyper-V manager. Open Query Analyzer on the VM, connect to the SQL Server instance, and restore the database. See [Appendix C](#) for details.



NOTE: It is important at this point to give your new database the same name as your original SQL Server database. Changing the name may break applications that refer to the database by name.

2. After restoring the database, but while still in Query Analyzer, reset the database access property to multi-user. See [Appendix C](#) for details.

Transferring logins

Windows and SQL Server logins

There are two methods of authenticating to SQL Server: Windows logins and SQL Server logins. Windows logins exist in the context of an Active Directory domain, and you can assign those logins rights to SQL Server resources. However, you create and manage SQL Server logins within SQL Server. While the processes for extracting login information and creating the transferred login entities on the consolidation server are very similar for both Windows authenticated logins and SQL Server authenticated logins,

you must take some extra steps to ensure a smooth migration for SQL Server logins.



NOTE: Take special care to avoid login issues for users when consolidating your databases, even in a virtual environment. Typically, these issues arise when a user had multiple SQL Server logins on multiple SQL Server instances in the previous environment. In that situation, a user may have used different passwords for different instances. However, depending on which databases you consolidate, the user may have only one login and associated password. You should note other issues in your planning documentation, such as permissions for multiple databases, default database, and language settings.

Below, we describe how to transfer logins using the Data Transformation Services (DTS) Transfer Logins Task. In SQL Server 2000, this procedure was the recommended approach provided by Microsoft. You should note, however, that Data Transformation Services (DTS) is a legacy technology in SQL Server 2000, and this approach is specific to moving logins between two SQL Server 2000 servers. For more information on this process and details, see <http://support.microsoft.com/kb/246133>.

See [Appendix D](#) for details on transferring logins.

To transfer logins, take the following steps on the new VM running SQL Server 2000. Note that steps 1 through 3 apply to both Windows logins and SQL Server logins.

- 1.** Open Enterprise Manager on the VM, and connect to SQL Server 2000. Browse to Data Transformation Services, right-click the task area, and choose New Package.
- 2.** Add the Transfer Logins task to the work area of the DTS package. Configure the Source server and the destination server, which in this case should be your virtual machine.
- 3.** After configuring the task, right-click the task, and choose Execute Step. DTS transfers the logins.

For SQL Server logins, you should ensure that you have properly mapped all database users on your virtual machine to SQL Server logins on that instance. To avoid having a database user who is "orphaned" from the login, use the provided stored procedure called `sp_change_users_login` to view orphaned users and to map

the user to the login. We provide an example of this process in [Appendix D](#).

Summing up

As this Guide has explained, the process of consolidating your SQL Server 2000 database servers to a virtual environment on a Dell PowerEdge Server running Hyper-V is relatively straightforward; you can perform a basic installation and move databases in a couple of hours. We still recommend, however, that you first invest ample time in the pre-consolidation planning phase, because doing so can help you avoid potential problems during your consolidation.

Appendix A. Creating the virtual machine and installing the guest OS

Creating the virtual machine

1. Click Start.
2. Expand Administrative Tools, and click Hyper-V Manager.
3. From the Action menu, select Connect to Server.
4. Accept the default of Local Computer, and click OK.
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. Accept the default location to store the Virtual machine, and click Next.
8. On the Assign Memory page, enter the amount of RAM you want to reserve for the virtual machine, in MB.
9. On the Configure Networking, choose Not Connected, and click Next.
10. Set an appropriate size on the Connect Virtual Hard Disk page. Accept the other defaults, and click Next. You will be adding other hard disks 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.

Adding processors and virtual hard drives to the virtual machine

Allow at least 10 minutes to add the virtual processors and hard drives.

Add any virtual processors, as well as each of the three non-operating system virtual hard drives to the VM, using the following steps:

1. In the Hyper-V manager window, right-click the virtual machine you just created, and choose Settings.
2. By default, Hyper-V creates the VM with one virtual processor. You can raise the number by clicking Processor in the left pane, then setting the Number of logical processors.



NOTE: Hyper-V supports four logical processors for Windows Server 2008 64-bit, but only two logical processors for Windows Server 2003 SP2 32-bit. When we set the number of processors for Windows Server 2003 SP2, Hyper-V offered four processors as an option. However, as this is not a Microsoft supported configuration, we did not experiment with it, nor do we recommend running with it.

3. In the left pane, click Add Hardware.
4. In the right pane, choose SCSI controller, and click Add.
5. In the left pane, choose SCSI controller.
6. In the right pane, select hard drive, and click Add.
7. In the right pane, click New.
8. In the Before You Begin page of the New Virtual Hard Disk Wizard, click Next.
9. On the Choose Disk Type page, choose Fixed size.
10. On the Specify Name and Location page, name the virtual hard disk file, and give the path name for the VHD file. Per best practices, store the virtual disks for the tempdb, logs, and user data on the appropriate drives. We put the relevant VHD file at the root of each drive we configured.
11. On the Configure Disk page, enter the size of the virtual disk. In our case, we used one-quarter the size of the physical RAID for each. Do not copy the contents of another disk. Click Next.
12. On the Completing the New Virtual hard Disk Wizard page, click Finish. (3 minutes per disk, although the time varied from disk to disk.)
13. After adding all three drives, click OK to close the Settings Window.



NOTE: You will notice that the Hyper-V manager created the first virtual disk on a virtual IDE controller. Do not try to change this. As per the Hyper-V documentation, the guest OS must be on a virtual IDE controller.

Installing the Windows Server 2003 SP2 guest OS

Allow 70 minutes to complete the installation of the Windows Server 2003 guest OS.

- 1.** In the Hyper-V manager, right-click the virtual machine, and choose connect. You will see a message that the virtual machine is turned off.
- 2.** On the Media menu, choose DVD Drive | Capture D:.
- 3.** Insert the Microsoft Windows Server 2003 Enterprise Edition CD into the DVD drive. From the Action menu, choose Start. You will see the blue Windows Setup screen as it loads. (2 minutes)
- 4.** On the Welcome to Setup screen, press Enter for a fresh install.
- 5.** Press F8 to accept the license agreement.
- 6.** Press Enter to install Windows on the unpartitioned space.
- 7.** Format the disk as NTFS. You may use the quick format if you choose.
- 8.** After the format, you will boot into the Windows Install. (12 minutes on the PowerEdge R900; 15 minutes on the PowerEdge R905)
- 9.** On the Regional and Languages Options screen, click Next.
- 10.** Enter your user name and organization name, and click Next.
- 11.** Enter your Product key, and click Next.
- 12.** Set the licensing mode as appropriate for your installation, and click Next.
- 13.** Enter a strong password for the administrator, and click Next.
- 14.** Set the date and time, and click Next. The install will continue. (14 minutes on the PowerEdge R900; 21 minutes on the PowerEdge R905)
- 15.** When the installer prompts you, log in using the password you set earlier. If necessary, use the Ctrl+Alt+Del button provided in the virtual machine interface to invoke the login screen.
- 16.** Insert the second Windows Server 2003 CD, and click OK.
- 17.** On the Windows Server 2003 R2 Setup Wizard, click Next.
- 18.** On the setup summary screen, click Next. (2 minutes)
- 19.** When the Setup Wizard is complete, click Finish.
- 20.** Before proceeding, you need to install SP2. We downloaded it and burned it to a CD.
- 21.** Put the SP2 CD in the DVD drive. Double-click the executable to run it. (3 minutes to load)
- 22.** On the SP2 installer splash screen, click Next.

- 23.**Accept the license agreement, and click Next.
- 24.**Click Next to accept the default uninstall folder.
- 25.**SP2 will start updating the VM. (8 minutes on the PowerEdge R900; 16 minutes on the PowerEdge R905)
- 26.**When SP2 finishes installing, restart the VM.
- 27.**Once the VM has rebooted, choose Insert Integration Services Setup Disk from the Action menu. Proceed with Integration Services installation.
- 28.**Turn off the VM. (2 minutes)
- 29.**Now that you have installed the Integration Services, in the Hyper-V manager, right-click the VM, and choose Settings.
- 30.**Click Network adaptor, and select your virtual NIC.
- 31.**Click Apply, and click OK.
- 32.**Right-click the VM, and choose Start.
- 33.**On the prompt to use Ctrl+Alt+Del to log in, click the three-button icon in the upper left corner of the virtual machine's window. If you press the Ctrl+Alt+Del keys, the host operating system will intercept the command.
- 34.**After restarting, configure your virtual NIC, using the steps in the section Setting IP address, subnet mask, and domain information below.

Setting IP address, subnet mask, and domain information on your VM

After rebooting the VM, set the static IP addresses and subnet mask on your VM using the following steps:

- 1.** Click Start | Control Panel | Network Connections | Local Area Connection.
- 2.** Click Properties.
- 3.** In the Local Area Connection Properties dialog, highlight Internet Protocol Version 4 (TCP/IPv4), and click Properties.
- 4.** Click the radio button next to Use the following IP address.
- 5.** Type the IP address you want the server to have, the Subnet mask, and the Preferred DNS server, and click OK.
- 6.** Close the Network Connections window.
- 7.** From the Start menu, right-click My Computer, and select Properties.
- 8.** Select the Computer Name tab.
- 9.** Click Change.

- 10.** Enter the server name.
- 11.** Click the Domain radio button.
- 12.** Enter the domain name.
- 13.** Enter the user name and password when the system prompts you to do so. (less than 1 minute)
- 14.** When the Computer Name/Domain Changes window appears, click OK to close it.
- 15.** Click OK to go past the warning to restart.
- 16.** Click OK to close the System Properties window.
- 17.** Click Restart Now to reboot the server. (3 minutes)

Configuring virtual data drives

Although we configured the data drives at the level of the host operating system, we have to repeat the process for the virtual drives. Configure each of the three non-operating system virtual hard drives on the server using the following steps:

- 1.** Click Start.
- 2.** Expand Administrative Tools, and click Computer management.
- 3.** In the left panel, expand Storage, and select Disk Management.
- 4.** If the disk is offline, right-click the disk name, and choose Online. If the Initialize Disk dialog box appears, click OK.
- 5.** Right-click the unallocated disk, and select New Simple Volume.
- 6.** On the Welcome to the New Simple Volume Wizard screen, click Next.
- 7.** Leave the volume size at the default, and click Next.
- 8.** Leave the drive letter at the default, and click Next.
- 9.** On the Format Partition screen, do the following:
 - a. Leave the File System as NTFS.
 - b. Set the Allocation Unit size to 64 K. (Note: Windows Server 2008 Enterprise Edition grays out the option to compress the drive.)
 - c. Fill in a volume name if you want the disk to have one.
 - d. Optionally, select the quick format checkbox.
- 10.** Click Next.
- 11.** On the Completing the New Simple Volume Wizard screen, click Finish.

Appendix B. Installing SQL Server 2000 Enterprise Edition

Install Microsoft SQL Server 2000 Enterprise Edition and SQL Server 2000 SP4 in the relevant virtual machine on the PowerEdge R900 or PowerEdge R905 server by following these steps. We provide approximate times for each group of steps in the first step of that group. Allow 15 minutes to install SQL Server 2000.



NOTE: We installed SQL Server 2000 Enterprise Edition using an MSDN distribution, and our steps below reflect that specific distribution.

Installing SQL Server 2000 with SP4

1. Start Hyper-V manager by clicking Start | Administrative Tools | Hyper-V manager.
2. In the Hyper-V manager, right-click the Windows Server 2003 virtual machine, and choose Connect. A message that the virtual machine is turned off appears.
3. On the Media menu, choose DVD Drive | Capture D:.
4. Insert the SQL Server 2000 CD into the DVD drive.
5. Choose Action, then Start. The Hyper-V splash screen appears, followed by the Windows Server 2003 splash screen.
6. Click the Ctrl+Alt+Del button in the virtual machine menu to log onto the VM. Provide the proper administrative credentials to log on.
7. Open Windows Explorer and browse to the path on the CD drive where the setup.bat file exists. On our MSDN distribution, this was at
D:\ENGLISH\SQL2000\ENT\
8. Double-click the file setup.bat.
9. A message appears warning that SQL Server 2000 SP2 and below is not supported by this version of Windows. Click Continue to proceed (we will install SP4 later).
10. On the Welcome screen, click Next to continue.
11. On the Computer Name screen, keep the default of Local Computer selected, and click Next.
12. On the Installation Selection screen, keep the default of Create a new instance of SQL Server, or install Client Tools selected, and click Next.

- 13.** On the User Information screen, enter your user information, and click Next.
- 14.** Click Yes to agree to the software license.
- 15.** On the Installation Definition screen, keep the default of Server and Client Tools selected, and click Next.
- 16.** On the Instance Name screen, keep Default Instance checked, which is the default action. Click Next.
- 17.** On the Setup Type screen, keep the Typical selection. Under Destination Folder/Data Files, click Browse to select your virtual hard drive location that you setup for SQL Server Data. Note that in SQL Server 2000, the installation routine places the data files and log files in this location. We show how to move the log files later. Click Next after choosing your data location.
- 18.** On the Services Accounts screen, keep the default selection to use the same account for each service, unless your needs require otherwise. Under Service Settings, we recommend using a domain user account. Enter in the appropriate credentials and domain, and then click Next.
- 19.** On the Authentication Mode screen, choose either Windows Authentication or mixed mode (mixed mode includes both SQL authentication and Windows authentication methods). We chose mixed mode.
- 20.** Assign a strong password to the "sa" user, and click Next.
- 21.** On the Start Copying Files screen, click Next to continue.
- 22.** On the Choose Licensing Mode screen, choose the appropriate licensing configuration for your situation, and click Continue.
- 23.** The installation now commences. (4 minutes)
- 24.** On the Setup complete screen, click Finish to close the setup program.
- 25.** To avoid any pending reboot issues with the service pack install, reboot the virtual machine by choosing Start | Shut down | Restart. (2 minutes)
- 26.** After restarting, log back into the virtual machine.
- 27.** Insert the CD containing SQL Server 2000 SP4.
- 28.** Ensure your CD drive is still capturing the physical DVD drive, by choosing Media | DVD Drive on the VM menu.

- 29.**In Windows Explorer, browse to the CD Drive, and locate the SP4 executable. Double-click it to begin SP4 installation.
- 30.**Click Next. The InstallShield Wizard extracts the files to the path you specified.
- 31.**Click Finish when extraction is complete. (2 minutes)
- 32.**In Windows Explorer, browse to the path you specified path where the InstallShield Wizard extracted the files.
- 33.**Double-click the setup.bat file to begin the SP4 installation.
- 34.**On the Welcome screen, click Next.
- 35.**Click Yes to agree to the software license agreement.
- 36.**On the Instance Name screen, click Next.
- 37.**Keep the default of using Windows account information, and click Next.
- 38.**Click Upgrade Microsoft Search and apply SQL Server 2000 SP4 (required). Click Continue.
- 39.**On the Error Reporting screen, click Next.
- 40.**On the Start Copying Files screen, click Next. The Service Pack installation commences. (2 minutes)
- 41.**You will receive a message regarding backup of the master and msdb databases. Click OK.
- 42.**On the Setup complete screen, click Finish.
- 43.**Restart the virtual machine to ensure all service pack updates are applied.

Moving tempdb files from the default location

The SQL Server 2000 installation routine stores tempdb by default on the on the C: drive, or wherever you specified during the installation routine with other user data. Our best practices recommend that you place tempdb on separate physical devices from user data for performance, unless your situation does not allow for this. Here, we show how to move the tempdb database and its associated files to another volume.

- 1.** In your virtual machine, click Start | All Programs | Microsoft SQL Server | Query Analyzer.
- 2.** Enter your virtual machine name, choose Windows Authentication, and click OK.
- 3.** Type in the following command to retrieve the metadata related to tempdb. Press F5 to execute the command.

```
sp_helpdb tempdb  
GO
```

4. Note the name column and the filename column.
5. Delete the previous command from the query window, and use the following commands to reconfigure the tempdb file locations. Note that the paths listed below are simply placeholders, and that you should modify them to fit your situation. Press F5 to execute these commands.

```
ALTER DATABASE tempdb MODIFY FILE ( NAME =  
tempdev,  
FILENAME = 'E:\newtempdbpath\tempdb.mdf' )  
GO  
ALTER DATABASE tempdb MODIFY FILE ( NAME =  
templog, FILENAME =  
'E:\newtempdbpath\templog.ldf' )  
GO
```

6. Restart the SQL Server service to make these changes permanent. Click Start | All Programs | Microsoft SQL Server | Service Manager.
7. Click the Stop button. Wait for the service to stop.
8. Click the Start button. Wait for the service to start.
9. Remove the original tempdb files, as the SQL Server instance has now recreated the tempdb files in the new location that you specified.

Appendix C. Migrating SQL Server 2000 databases from your physical machine to your virtual machine

In this appendix, we describe how to move your SQL Server 2000 databases from a physical server to a virtual server. Note that moving SQL Server 2000 full text catalogs, and other SQL Server 2000 products such as Analysis Services and Reporting Services, is outside the scope of this Guide.

1. On your physical SQL Server 2000 server, log into Windows with sufficient credentials.
2. Click Start | All Programs | Microsoft SQL Server | Query Analyzer.
3. Enter your physical machine name, choose Windows Authentication, and click OK.
4. Set your database to single user mode to ensure no updates occur during the consolidation process. Here, we use the sample database in SQL Server 2000 called Northwind. Press F5 to execute this command.

```
ALTER DATABASE Northwind SET SINGLE_USER  
GO
```

5. Type the following command to back up the relevant database. Be sure you have enough disk space in your backup location. Press F5 to execute the command.

```
BACKUP DATABASE Northwind TO DISK =  
'C:\backuplocation\Northwindbackup.bak'  
GO
```

6. Now on your virtual machine, log into Windows with sufficient credentials.
7. Copy the backup file to your virtual machine. We used simple file sharing, but you could use external hard drives, etc. for a more manual file transfer.
8. Start Query Analyzer on the virtual machine by choosing Start | All Programs | Microsoft SQL Server | Query Analyzer.
9. Use the following command to retrieve metadata from the backup file. This does not restore the database, but allows you to view metadata inside the

backup file. Note the LogicalName field. Also, note that the path in the script below is a placeholder, and that you should adjust it for your situation. Press F5 to execute the command.

```
RESTORE FILELISTONLY
FROM DISK =
'E:\pathtobackupfile\Northwindbackup.bak'
GO
```

- 10.** Use the following command to restore the database to your virtual machine. We also include the WITH MOVE clause because there is a good possibility that, as you move from physical to virtual environments, your drive letters, data locations, and log locations will change. Press F5 to execute the command.

```
RESTORE DATABASE Northwind FROM DISK =
'C:\backuplocation\Northwindbackup.bak'
WITH MOVE 'Northwind' TO
'E:\pathtosqldata\Northwind.mdf', MOVE
'Northwind_log' TO
'F:\pathtosqllogs\Northwind_log.ldf'
GO
```

- 11.** Set the database back to multi user mode. Use the following command. Press F5 to execute the command.

```
ALTER DATABASE Northwind SET MULTI_USER
GO
```

Appendix D. Transferring logins

In this appendix, we provide detailed instructions for transferring logins from a physical server running SQL Server 2000 to a virtual machine running SQL Server 2000. This procedure is the approach Microsoft recommends. You should note, however, that Data Transformation Services (DTS) is a legacy technology in SQL Server 2000, and that this approach is specific to moving logins between two SQL Server 2000 servers. For more information on this process and details, see <http://support.microsoft.com/kb/246133>.

- 1.** On the virtual machine, start Enterprise Manager by clicking Start | All Programs | Microsoft SQL Server | Enterprise Manager.
- 2.** If you have not registered any SQL Servers in Enterprise Manager, you must register them.
- 3.** Expand Microsoft SQL Servers, then right-click SQL Server Group, and choose New SQL Server Registration.
- 4.** Click Next.
- 5.** Choose the virtual machine and the physical machine from which you are transferring logins. Click Add.
- 6.** Click Next.
- 7.** Connect using Windows authentication if possible.
- 8.** Choose to add the SQL Servers to the default SQL Server group.
- 9.** Click Finish, and click Close.
- 10.** Click the plus sign to show your virtual machine, then right-click Data Transformation Services, and choose New Package.
- 11.** Click Task | Transfer Logins task.
- 12.** On the Source tab, type in the name of your physical SQL Server 2000 server.
- 13.** On the Destination tab, type the name of the virtual SQL Server machine.
- 14.** On the Logins tab, choose the applicable option regarding which logins you would like to transfer. We chose to keep the default.
- 15.** Click OK.
- 16.** Right-click the Transfer Logins task icon, and choose Execute Step. You will receive a message saying that the step successfully executed. Click OK.

17. Check the logins in SQL Server Enterprise Manager on the virtual machine to ensure that the DTS package successfully executed.

Mapping transferred logins to database users

After transferring SQL Server logins from physical server to virtual server, you may need to map those logins to the database users that you migrated during the restore process. This process assumes that you have already migrated the applicable database to the new server.

1. On the virtual machine running SQL Server, log into Windows.
2. Click Start | All Programs | Microsoft SQL Server | Query Analyzer.
3. Enter your virtual machine name, choose Windows Authentication, and click OK.
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 performing the following steps.
6. For each login you wish to map, enter the following command, and press F5 to execute:

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

7. 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. Upon successful execution of this procedure, you will see output similar to the following:

```
The number of orphaned users fixed by  
updating users was 1.
```

About Principled Technologies



Principled Technologies, Inc.
1007 Slater Road, Suite 250
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 you assess how it will fare against its competition, its performance, whether it's ready to go to market, and its quality and reliability.

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

Principled Technologies 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.
