Microsoft

2

Second Edition

Updated for Service Pack One and R2

Microsoft® WINDOWS SERVER™ 2003

Administrator's Companion

Charlie Russel
Sharon Crawford
Jason Gerend

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Library of Congress Control Number 2005936849

Printed and bound in the United States of America.

1 2 3 4 5 6 7 8 9 QWT 0 9 8 7 6 5

Distributed in Canada by H.B. Fenn and Company Ltd.

A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office or contact Microsoft Press International directly at fax (425) 936-7329. Visit our Web site at www.microsoft.com/mspress. Send comments to mspinput@microsoft.com.

Acquisitions Editor: Martin DelRe
Project Editor: Karen Szall
Technical Editor: Mitch Tulloch
Copy Editor: Roger LeBlanc
Production: Elizabeth Hansford
Indexer: Julie Hatley

Body Part No. X11-97525

Contents at a Glance

Part IV
Security

Part V
Support Services and Features

Part VI
Internet Servers and Services

Part VII
Tuning, Maintenance, and Repair

Part VIII

Appendixes

Table of Contents

Part II
Installation and Configuration

Part III
Network Administration

Part V

Support Services and Features

Part VII

Tuning, Maintenance, and Repair

Chapter 5

Getting Started

The first four chapters of the book have dealt with abstract planning. This chapter delves into the specific details of planning and implementing a deployment of Microsoft Windows Server 2003 R2, or any other recent version of the Microsoft Windows operating system.

Installing Windows manually is a time-consuming task on a single system; deploying it to a group of computers is a mind-numbing experience unless you automate the process. Fortunately, there are numerous ways to automate the deployment of Windows. These methods range from using simple answer files that run Windows Setup in unattended mode from CD-ROM on a single system to methods that deploy disk images of completely configured systems over the network to any number of target computers.

This chapter discusses how to design a deployment environment suitable to your organization's needs, how to create an automated deployment environment (if you choose to automate the deployment process), and how to install Windows manually or using an automated method.

Designing a Deployment Environment

It is helpful to design a deployment environment that allows you to deploy Windows efficiently to the client and server systems in your organization. This section discusses how to design a deployment environment that provides the deployment speed and control you need while maintaining an acceptable level of complexity—regardless of whether you want to manually install Windows on a single system or mass deploy it to thousands of computers.

Under the Hood Setup in Windows Server 2003 R2 and Windows XP Tablet PC Edition 2005

Windows Server 2003 R2 and Windows XP Tablet PC Edition 2005 include two CDs. The first CD is a normal Windows Server 2003 with Service Pack 1 CD (in the case of Windows Server 2003 R2), or a normal Windows XP Professional with Service Pack 2 CD (in the case of Windows XP Tablet PC Edition 2005). The second CD contains a \Cmpnents folder that holds the files that distinguish Windows Server 2003 R2 or Windows XP Tablet PC Edition 2005 from a normal version of Windows Server 2003 or Windows XP Professional.

If you use a Windows Server 2003 R2 product key during installation, Windows Server 2003 with Service Pack 1 Setup installs Windows Server 2003 R2 (but no optional components) after Windows Setup (Winnt32.exe or Winnt.exe) is complete by launching Windows Server 2003 R2 Setup (Setup2.exe) from the \Cmpnents\R2 folder of the second CD. If you use a Windows XP Tablet PC Edition 2005 product key during installation, Windows XP Professional with Service Pack 2 Setup installs Windows XP Tablet PC Edition 2005 by installing additional files from the Cmpnents\TabletPC folder on the second CD.

This affects you in four ways:

- You need to use the correct Windows Server 2003 R2 or Windows XP Tablet PC Edition 2005 product key.
- When performing a manual installation, Windows Setup (Winnt32.exe or Winnt.exe) prompts you to insert the second CD during Setup (for Windows XP Tablet PC Edition 2005) the first time an administrator logs in after Setup completes (for Windows Server 2003 R2). You can click Cancel if you want the server to run Windows Server 2003 with Service Pack 1 instead of Windows Server 2003 R2.
- When performing an automated installation of Windows Server 2003 R2, you must include the \Cmpnents folder in your installation source, and include a reference to Setup2.exe in the answer file. For more information, see the "Using Unattended Setup with Windows Server 2003 R2" section of this chapter.
- To install optional Windows Server 2003 R2 components such as the new File Server Management console, you must be able to access the \Cmpnents folder and \i386 or \amd64 folders, either locally, on a network share, or on the Windows Server 2003 R2 Disc 2.

Choosing an Installation Method

Windows installation methods fall into one of two broad categories—Setup-based methods and image-based methods. Setup-based methods use Windows Setup (Winnt32.exe or Winnt.exe) to install Windows on the target computer. Image-based methods copy a disk image of a reference computer with Windows already installed on it to the target computer, which must use hardware that is similar to the reference computer.

Choosing a Setup-Based Installation Method

Setup-based installation methods are relatively simple to use, and you can use them on any hardware configuration, making them ideal for organizations with heterogeneous hardware configurations. The following list describes the three different Setup-based installation methods. Table 5-1 lists the key attributes of each method.

- **Manual installation** Performing a manual installation from the Windows CD-ROM or a network share is the simplest way to install Microsoft Windows. Use this method when installing on a single computer and none of the other installation methods are convenient.
- **Unattended installation using an answer file** This method automates the installation of Windows from a CD-ROM, local source, or network share (called a *distribution share*) by using a text file called an *answer file* that provides the answers to each question in Windows Setup. Use an answer file to automate Windows Setup and exert more control over Windows default settings than can be provided by a manual installation.
- **Remote Installation Services (RIS)** A Windows server service that enables users to boot a computer over the network from the RIS server and automatically run Windows Setup in unattended mode. Use RIS to automate Setup on any computer that supports booting from the network via Preboot Execution Environment (PXE); to allow users to easily install or reinstall Windows; and to provide central control over the Setup process. For more information about RIS, see Chapter 28.

Note You can increase the number of simultaneous clients that the server hosting the distribution share can handle by copying the installation files from the distribution folder to the target computer's hard drive and then running Setup locally instead of across the network. Although copying the installation files puts a greater load on the server at first, after Windows finishes downloading the files, no additional load is placed on the server, freeing it to service other clients.

Table 5-1 Manual setup, unattended setup, and RIS attributes

Attribute	Manual Setup	Unattended Setup	RIS
Speed	Slow	Medium	Medium
Boot method	Windows CD or Windows operating system	Windows CD, Windows operating system, or boot floppy	PXE boot from network, or network boot using RIS boot disk
Install source	CD, file share, or hard drive	CD, file share, or hard drive	RIS Server
Supported operating systems	All versions	All versions	Windows Server 2003*; Windows XP Professional*; Windows 2000
Hardware similarity requirements	None	None	None
Maximum simultaneous clients	1	Scales with hardware	75 per server
Configuration information	Manual entry	Unattend.txt	Active Directory and *.sif answer file

* RIS also supports Windows XP Tablet PC Edition and x64 editions of both Windows Server 2003 and Windows XP Professional.

Choosing an Image-Based Installation Method

Image-based installation methods are the fastest methods of installing Microsoft Windows; however, they work best when there is a high degree of hardware similarity between the target computer and the reference computer from which you created the disk image. Too much variation between systems can lead to a multitude of disk images or longer install times when installing additional drivers during Sysprep Mini-Setup. (Sysprep Mini-Setup is an abbreviated version of Windows Setup that runs on the first boot after using Sysprep to reseal a computer for disk imaging.) Finding and maintaining the proper drivers for a disparate set of systems can also be a challenge, which is why many organizations deploy only a few model lines of computers from the same manufacturer.

These characteristics make image-based installation methods ideal for organizations that need to deploy the Windows operating system and applications to large numbers of similar

systems that use the same hardware abstraction layer (HAL). The following list discusses the image-based setup methods that are currently available; Table 5-2 lists the key attributes of each method.

- **Sysprep with disk imaging** Creates an exact copy or "image" of a Windows installation, which can then be copied onto the hard disk of another computer with similar hardware. Use disk imaging in conjunction with the Sysprep tool to rapidly deploy Windows and applications to a large number of computers that use the same HAL and similar hardware devices.
- **RIS with Remote Installation Preparation (RIPrep) images** Boots a computer over the network from the RIS server and automatically installs a disk image. This Windows server service starts the target computer across the network, downloads a RIPrep disk image from the RIS server, and performs a full Plug and Play (PnP) detection pass. Use RIS with RIPrep images to quickly deploy Windows and applications to any computer that supports starting from the network via Preboot Execution Environment (PXE). For more information about RIPrep, see Chapter 28.
- **Automated Deployment Services (ADS)** Enables administrators to rapidly deploy Windows Server in a data center. This Windows server service allows the administrator to remotely start a computer over the network and install or reinstall a Windows Server disk image on the computer. Use ADS to centrally manage the deployment of Windows Server in a data center. For more information about ADS, see the Microsoft Web site at *http://www.microsoft.com/windowsserver2003/technologies/management/ads/default.mspx*.
- **SMS 2003 Operating System Deployment Feature Pack** Enables administrators to centrally manage the deployment of client and server disk images on a network. Using Systems Management Server (SMS), you can advertise an operating system package that SMS installs automatically on targeted systems that already have the SMS client installed. You can also install the operating system package by starting Windows Preinstallation Environment (Windows PE) and connecting to the SMS server. (Windows PE is included with the SMS 2003 Operating System Deployment Feature Pack.) Use the SMS 2003 Operating System Deployment Feature Pack to centrally manage the deployment of Windows on a network that uses SMS to manage systems. For more information about SMS, see the Microsoft Web site at *http://www.microsoft.com/smserver/*.

Table 5-2 RIPrep, ADS, and SMS attributes

Attribute	Sysprep and Disk Imaging	RIPrep	ADS	SMS
Speed	Fastest	Fast	Fastest	Fastest
Boot method	Windows operating system, Windows PE, or disk-imaging boot floppy	PXE boot from network, or network boot using RIS boot disk	PXE boot from network	Windows operating system, or Windows PE via RIS or CD-ROM
Install source	Disk image on network share or CD	Disk image on RIS server	Disk image on ADS server	SMS package on SMS server or CD-ROM
Supported operating systems	All versions	Windows Server 2003*; Windows XP Professional*; Windows 2000	Windows Server 2003; Windows 2000 Server with Service Pack 3 or later	Windows Server 2003; Windows XP; Windows 2000
Hardware similarity requirements with reference computer	Same HAL (alternative mass storage drivers must be added manually)	Same HAL	Same HAL (alternative mass storage drivers must be added manually)	Same HAL (alternative mass storage drivers must be added manually)
Maximum simultaneous clients	Scales with hardware	75 per server	128 per server	Dependent on SMS infrastructure
Configuration information	Disk image and Sysprep.inf answer file	Disk image, Active Directory, and *.sif answer file	Disk image and Microsoft SQL Server database or Microsoft SQL Server Desktop Edition (MSDE) database	Disk image and Microsoft SQL Server database

* RIPrep also supports Windows XP Tablet PC Edition and x64 editions of both Windows Server 2003 and Windows XP Professional.

Note You can add support for mass storage devices not included in a reference image by adding the drivers to the Sysprep folder on the target system and using the SysprepMassStorage section of Sysprep.inf. For more information, see the *Microsoft Windows Corporate Deployment Tools User's Guide*.

Choosing a Preinstallation Environment

Before you can install Microsoft Windows, you must boot the computer into a suitable operating system (referred to as the *preinstallation environment*). You can use any of the following methods of booting a computer to install Windows:

- Boot from a Windows CD-ROM directly into Windows Setup.
- Boot from a Windows 98– or MS-DOS-based boot disk, and launch 16-bit Windows Setup (Winnt.exe) from a local or network source.
- Boot from an existing operating system such as a "safe OS" installation of Windows created for installation and recovery purposes, and launch 32-bit Windows Setup (Winnt32.exe) from a local or network source.
- Perform a Preinstallation Execution Environment (PXE) boot from the network into Windows Setup via RIS or ADS (or use a RIS boot disk to start from the network).
- Boot Windows PE from a CD-ROM or RIS server and then launch 32-bit Windows Setup (Winnt32.exe) from a local or network source, copy a disk image from a file share, or connect to an SMS server with the SMS 2003 Operating System Deployment Feature Pack.

Important Each operating system installation that belongs to a workgroup or Windows domain must have a unique computer name, even if the installations are on the same computer in a dual-boot or a virtual machine configuration.

Choosing a Software Update Solution

To keep your network secure, you must update every Windows installation on your network in a timely manner with the latest software updates. This is especially true for fresh installations of Windows versions earlier than Windows XP with Service Pack 2 or Windows Server 2003 with Service Pack 1; earlier versions of Windows are potential targets for viruses before you can install antivirus software and the latest software updates, even if you enable the Internet Connection Firewall.

This vulnerability highlights the need to select a software update solution when you design your Windows deployment environment, if you have not already. Selecting a software update solution before you create your deployment environment makes it easier for you to integrate important software updates into your deployment process.

There are a number of software update solutions that you can use to install the latest software updates on computers running Microsoft Windows:

- Install updates manually using Microsoft Update or Windows Update.
- Install updates automatically using Automatic Updates, with or without Windows Server Update Services (WSUS) for central control.
- Install updates automatically using SMS for central control.
- Use a third-party patch management program.
- Add updates to the Windows installation process.

Most organizations use more than one of these methods. For example, a common practice when installing Microsoft Windows is to install Windows from a distribution share or disk image with the latest service pack, and script the installation of any critical software updates into the end of the installation process using an answer file. After Setup completes, you can manually run Microsoft Update and enable Automatic Updates, or use SMS or a third-party patch management program to perform future software updates.

For more information on managing software updates, see Chapter 28.

Choosing an Application Deployment Solution

A computer is of minimal value in an organization unless it has the applications the user of the computer needs to perform her or his job. You can install applications, drivers, and services during the Windows installation process, or you can install them afterwards. The following list describes a number of methods that you can use to deploy applications:

- **Manual installation** Performing a manual installation is the simplest way to install an application. Use this method when installing on a single computer and none of the other installation methods are convenient.
- **Group Policy** Using Group Policy to deploy applications is an efficient way to deploy applications to a large number of computers in an Active Directory environment. To install applications automatically using Group Policy, the applications must use Microsoft Installer .MSI packages; otherwise, you can publish any application setup program for manual or semi-automated installation by the user. Use this method when installing applications on many computers in an Active Directory environment, when central management capability is important, and when initial setup time, scheduled installations, managed bandwidth and error reporting are not critical. See Chapter 28 for more information about using Group Policy to deploy applications.

- **SMS** Using SMS to deploy applications provides the highest level of control and reporting when deploying applications to a large number of computers. Use this method on large, complex networks, or when you need the highest level of control over the deployment process and the added complexity and cost of SMS is acceptable.
- **Install or stage applications during Windows Setup** Installing applications via a silent install immediately following Windows Setup, or staging applications by copying the installation files to the target computer's hard drive for installation later, is a simple method of quickly making applications available to users. Use this method with applications that support unattended installation, and when quick and simple unattended installation of Windows and key applications is more important than central management ability.
- **Install applications prior to disk imaging** Installing applications on the reference computer before using disk-imaging software to duplicate it is the fastest way to deploy a common set of applications. Use this method when you want to minimize the initial setup time for a common set of applications. If you want to manage these applications centrally, install the applications using your chosen application deployment method (for example, SMS).

Understanding Licensing and Product Activation

There are two licensing issues that administrators must understand with Windows Server 2003—licensing modes and product activation.

Licensing Modes

There are two types of licensing modes for Windows servers: Per Server licensing and Per Device Or Per User licensing (formerly called Per Seat). With Per Device Or Per User licensing, each client that accesses the server needs its own Client Access License (CAL). You can purchase CALs for users, for devices, or for a mix of the two. Purchase user CALs for users who connect to servers with more than one device (for example, a user with a desktop, a laptop, and a PDA), and purchase device CALs for devices that are used by more than one user (for example, a kiosk computer). Clients with a CAL can connect to any number of servers, making this method the most common licensing method for companies with more than one Windows server.

Per Server licensing requires the server to have a CAL for each concurrent connection. For example, if you choose the Per Server licensing mode with 50 concurrent connections, the server can support a maximum of 50 simultaneous client connections. This licensing mode works well for companies that use a single Windows server.

If you are unsure which licensing mode to use, visit the Microsoft Web site at *http://www.microsoft.com/windowsserver2003/howtobuy/licensing/default.mspx* or contact your local Value Added Reseller (VAR) or Microsoft account representative. You can also choose Per Server and switch later to Per Device Or Per User if you made a mistake. You can switch from Per Server to Per Device Or Per User once (without additional cost), but not from Per Device Or Per User to Per Server.

Note In Control Panel, click the License icon to keep track of the license purchases and holdings, or to switch licensing modes.

Real World Licensing for Terminal Servers and External Users

You must use Per Device Or Per User mode on Terminal Servers, and you must purchase additional Terminal Services CALs for users or devices (though these users or devices do not require a separate Windows CAL unless they also access the server from Windows). Windows 2000 users cannot access Windows Server 2003 Terminal Servers without a CAL, unlike with Windows 2000 Server.

There are three types of Terminal Services CALs: a TS Device CAL, a TS User CAL, and a TS External Connector CAL (for qualified external users who the organization does not employ in any way). For more information about Terminal Server licensing, see the Microsoft Web site at *http://www.microsoft.com/windowsserver2003/howtobuy/licensing/ts2003.mspx*.

All users that authenticate with the server in any way (or access the server after authenticating) require a CAL—even authenticated Web site users (except in some cases when using Windows Server 2003 Web Edition). You can purchase an External Connector License for servers that external users access. For example, if a Web site user logs on to a front-end Web site for a SharePoint site hosted on another server, you need either a CAL for the user, or an External Connector license for the Web server, *and* the SharePoint server. The External Connector License replaces the Windows 2000 Server Internet Connector License.

Product Activation

When you install Windows Server 2003, Windows XP, or a variety of other Microsoft programs from retail media, you have 30 days to activate the product. After the grace period, you cannot log on until the product is properly activated (though the computer will run until you restart it). If you change the hardware enough to require activation (which usually entails a motherboard change plus a couple of other devices), Windows gives you three days before it stops working (unless you are using Windows XP without any service packs, in which case you must activate the product immediately).

This is a real pain if you mass deploy systems, because you need a different product key for every computer and must activate each computer separately. Owning a license for each copy is not enough.

To eliminate the hassle of product activation, purchase volume licenses for Windows and Microsoft Office, because copies of Windows and Office bought in volume do not require product activation. Note that volume licenses are less expensive than retail copies, and they are available for as few as five total Office and Windows products—for example, two copies of Windows and three copies of Office. Some volume license agreements also give you access to Windows PE—an invaluable tool for deploying Windows.

More Info For more information about volume licensing, see the Microsoft Licensing Web site at *http://www.microsoft.com/licensing*.

Designing a Test Lab

A test lab provides a safe location in the organization to create and test Windows installations before deploying them to the network at large, and it is an integral component of a medium- or large-sized Windows deployment.

Large organizations that do not already have a test lab should consider creating a dedicated network for testing that closely matches the production network. You can use the deployment test lab for testing a variety of Windows deployment and management scenarios, such as software updates and line of business applications, or you can create separate labs for each of these testing tasks.

Smaller organizations can often use test systems on the production network or use a Microsoft Virtual PC or Microsoft Virtual Server environment for testing. However, when using a virtual test environment it is important to validate against real hardware that is representative of the production environment. You cannot test display card drivers on a virtual machine, for example.

An ideal test lab has the following characteristics:

- Contains test computers that closely represent the computers in the production environment to which you want to deploy Windows.
- Uses the same server configurations as the production network.
- Mirrors the production network environment with respect to network topology, devices, and settings.
- If load testing a server is important, uses similar hardware configurations for the test server and production servers, and provides a method of simulating the typical and maximum loads you expect the servers to encounter during deployment.

- Is isolated from the production network sufficiently to prevent the lab from affecting the running network.
- Is well documented and easy to roll back to the reference state after testing a change.

More Info For more information about creating test labs, see "Planning, Testing, and Piloting Deployment Projects" in the Windows Server 2003 Deployment Guide at *http://www.microsoft.com/resources/documentation/WindowsServ/2003/all/deployguide/en-us/Default.asp*.

Planning Server Configurations

If the deployment method you select makes use of one or more servers, it is important to plan the server configurations to provide sufficient performance and support the necessary services for the deployment. You must also plan the configurations of any servers that you are deploying to your production network.

Planning Server Roles

The server role has a major impact on the rest of the server configuration. Decide what role a server will play before planning other aspects of the server configuration. Then assess the server's system requirements, partition layout, and security needs accordingly. Heavily used database servers, for example, usually require extra random access memory (RAM) and fast disk subsystems. RIS servers require multiple partitions, and Web servers have the strictest security requirements.

When creating a deployment environment for Windows, ensure that you fill the following server roles on the deployment network:

- **File server** If you are using a distribution share to install Windows or store disk images, you need a file server with sufficient capacity to store all files and provide satisfactory performance for the peak number of simultaneous clients you expect to deploy.

  You can also use multiple servers to service clients and replicate the distribution share (or other shares) between the servers using Distributed File System (DFS) or the Windows Server 2003 R2 Distributed File Replication Service (DFRS).

- **Domain controller** If you are using an Active Directory environment for the deployment environment, you need at least one domain controller to service logon requests.

- **DHCP server** You need one or more DHCP servers to provide dynamic IP addresses to client systems, both during and after deployment, unless you choose to use static IP addresses (which is not recommended).
- **DNS server** One or more DNS servers are required to provide adequate Active Directory functionality and make name resolution more convenient.
- **WINS server** If your preinstallation environment is MS-DOS based and you have multiple subnets, you need a WINS server to ensure proper NetBIOS name resolution.
- **RIS, ADS, and SMS servers** If you are using RIS, ADS, or SMS to deploy Windows, set up these servers and size them properly to handle the anticipated load.
- **SQL Servers** If your deployment environment stores configuration data in a SQL Server database, set up these servers and size them properly to handle the anticipated load.

Note Take a few moments to think about naming conventions before you commit yourself to a naming scheme. Sometimes system administrators devise arbitrary schemes based on algorithms known only to them, or they attempt to insert charm into the process of computer naming. Block those impulses! It is easy for *you* to keep a map of what and where the different clients and servers are on the network, but if you make life hard on users, you will pay in the end.

Note Usually, short names that indicate the function of the computer or the owner of the computer (for client systems) are best. For example, "Legal_Files" is a good name for a file server in the Legal department.

Assessing System Requirements

Make sure that the servers are powerful enough to handle the anticipated load during deployment while providing an adequate response time.

Table 5-3 lists the minimum system requirements for Windows Server 2003 along with some more practical recommendations for the minimum necessary hardware. Before you buy server hardware, check the Windows Server Catalog on the Microsoft Web site (*http://www.microsoft.com/windows/catalog/server*).

Note The late Duchess of Windsor was fond of saying that you can never be too rich or too thin. Now you can add the axiom that you can never have too much processing power, RAM, or hard disk space on a server. The only restriction is economic. Get the most powerful server you can afford.

Table 5-3 Minimum requirements for achieving adequate performance

Minimum	Recommended Minimum
x86: Intel Pentium 133 x64: 1.4-GHz processor	One or more Intel Pentium III 1-GHz or faster microprocessors (or compatible processors)
x86: 128 MB of RAM x64: 512 MB of RAM	512 MB of RAM or more
VGA monitor	Super VGA monitor capable of at least 800-by-600 resolution
Keyboard and mouse or other pointing device	Keyboard and mouse or other pointing device
x86: 1.5 GB of free hard disk space x64: 4 GB of free hard disk space	10 GB of free disk space on a 7200-rpm or faster hard disk
Bootable CD-ROM	Bootable 12x or faster CD-ROM or DVD-ROM drive (El Torito–compatible)
No floppy disk	1.44-MB floppy disk
One or more network adapters	One or more Peripheral Component Interconnect (PCI), PCI-X, or PCI Express–based Fast Ethernet or Gigabit Ethernet network adapters with PXE support

Note Yes, the name of the specification is indeed El Torito. The engineers who developed the standard named it after the restaurant where they had held many, presumably happy, collaborative meetings.

Planning Partitions

Before installing Windows, determine how you want to partition the hard drives. A single partition works nicely for client installations but is generally unsuitable for servers. One common practice is to create a 6- to 10-GB partition for the operating system, and another partition with the remaining space. In this way, the operating system is separate from applications and data (especially the Internet Information Services \Inetpub folder), and you can install services that cannot be installed on the system partition, such as RIS. Creating a 6- to 10-GB system partition and a second data and application partition also works well for most servers that use a redundant array of independent disks (RAID) because a hardware RAID appears to the operating system as a single drive.

Note Some administrators like to place the Windows page file on a dedicated partition to reduce page file fragmentation; however, there are easier ways to prevent page file fragmentation, as discussed in Chapter 7.

Windows Setup does not support dynamic volumes well, which can lead to problems when recovering from a serious error or upgrading the operating system. For this reason,

when using dynamic volumes (including software RAID sets), consider leaving the system partition a basic disk and creating dynamic volumes for data and applications only. You can also install a "safe OS" copy of Windows on a basic partition—you can use the "safe OS" copy to restore from backup in case of a disaster. If you install a "safe OS" (or parallel installation, usually of Windows XP), it is important to lock down this installation, because you cannot install software updates without restarting into the installation. You also need a separate license and computer name for this installation.

Note You cannot perform a clean install on a dynamic volume unless the dynamic volume is "hard-linked." Only dynamic volumes that you upgraded from basic volumes are hard-linked.

Real World Just Say "NTFS"

Although FAT16 and FAT32 had their places in the past, there is no place for them anymore on the hard drives of today's Windows clients and servers. Simply put, NTFS is more reliable, secure, and efficient than FAT and FAT32. With Windows Server 2003, NTFS is also every bit as fast. (FAT previously maintained a slight speed advantage for some tasks.)

If you absolutely must use a legacy version of Microsoft Windows 95, Windows 98, or Windows Me, you *can* use FAT partitions to perform a dual boot. However, a more powerful, not to mention more secure, solution is to use Microsoft Virtual PC or Virtual Server to run these operating systems in a virtual machine on a system running Windows XP or Windows Server 2003.

Planning Server Security

It is important to ensure adequate security for the servers in the deployment environment as well as the servers you deploy into the production network. Take the following measures to secure the deployment network:

- Create a dedicated user account just for installations that is a member of a new group named Install Users (or something similar). Limit the permissions given to the account, and consider setting the NTFS permissions for this account to Deny on all folders except the distribution share and subfolders.

  To deny the account dial-in rights, right-click the account in Active Directory Users And Computers, click Properties, click the Dial-In tab, and then select the Deny Access check box.

- Set the permissions on the distribution share to grant Read and Execute permissions to the Install Users group. Check the NTFS permissions on your folders to ensure that members of the Install Users group do not have access to any other folders.

To secure the servers you deploy, follow these recommendations:

- Unplug the server from the Internet during Windows Setup, even if it is destined for life as a Web server.
- Install Windows to an NTFS partition.
- Create a strong local administrator password during Setup. (Microsoft Windows Server 2003 prompts you automatically for a different password if the one you choose is too simple.) One common practice is to join the computer to a Windows domain, set a random local Administrator password, and then use the domain administrator accounts for all administration tasks.
- Carefully guard any answer files that contain user names and passwords (as described in the "Using Setup Manager" section of this chapter).
- Physically secure the computer as appropriate for its role and contents:
  - Place servers in a locked server room. Give the key or combination only to people with a demonstrated need for it. Create a system that tracks who enters the room and when.
  - Use case locks and do not leave the keys in them.
  - Remove the floppy disk drive if it is not necessary; otherwise, consider floppy disk locks.
  - After you install Windows, set the boot order in the Basic Input Output System (BIOS) to boot only from the hard drive. This prevents someone from using special boot disks or CD-ROMs to access the contents of the hard drive and reset the local administrator password.
  - Set a BIOS password to prevent unauthorized access to the BIOS.
  - Change the operating system selection timeout period to 0 so that Windows boots automatically.

Security Alert All versions of Microsoft Windows prior to Windows XP with Service Pack 2 or Windows Server 2003 with Service Pack 1 are vulnerable to viruses during the period between the completion of Setup and the installation of antivirus software, even if you enable the Internet Connection Firewall. This can be a problem even if the computer is behind a corporate firewall because many private networks have viruses circulating on them. The new Windows Firewall can protect a system until you can install antivirus software and the latest software updates.

Real World Password Security

To make the system as secure as possible, always assign a password to the administrator account, preferably a password at least seven characters long and consisting of mixed letters and special characters, uppercase letters, and lowercase letters. Use acronyms for phrases that are meaningful to you, easy to remember, and unlikely to be meaningful or memorable to anyone else, such as Uk,Ur?Ue! (which stands for "You know, you are what you eat!").

Clear the logon history after installing Windows so that would-be hackers must figure out both the password and the user name. To do this on a standalone server, click Start, choose Administrative Tools, and then choose Local Security Policy. Select Local Policies, select Security Options, double-click Interactive Logon: Do Not Display Last User Name, choose Enabled, and then click OK. Use Group Policy to control this on a member server or domain controller.

Because it is possible to disable any administrator account, including the built-in Administrator account, it is wise to have a backup account. Use the built-in Administrator account to make a second account with full administrative privileges, stash the password and user name somewhere safe, and then relegate that account to semiretirement. For extra credit, rename the built-in Administrator account and then create a decoy account named Administrator. Give the decoy account no permissions and disable it. The hackers can pound away at this account as long as they like but it won't do them any good.

Creating Your Deployment Plan

Before you create your deployment environment and begin the deployment process, it is important to create a deployment plan. This is a document or series of documents that serves as a road map for the deployment process. The deployment plan should include the following items:

- **A budget** Determine the level of funding available for the project, and keep it up to date as the project progresses.
- **A schedule** Create a timeline of when to perform each phase of the deployment, and update it as you achieve (or delay) each milestone.
- **A test plan** Document specific steps that you can perform to verify that an installation performed in your test lab or in a pilot deployment is fully functional.

- **List of configurations** Create a list of configurations that you need to deploy and the answer files or disk images that support these configurations.
- **Deployment steps** Document the exact steps that administrators must perform to deploy Windows to your network. Use these steps as the basis for the test plan, and update it with results from testing and pilot deployments.

More Info For information about performing a pilot deployment, see "Designing a Pilot Project" in the Windows Server 2003 Deployment Guide at *http://www.microsoft.com/technet/prodtechnol/windowsserver2003/library/DepKit/d2ff1315-1712-48e4-acdc-8cae1b593eb1.mspx*.

Creating Your Deployment Environment

This section discusses how to perform common tasks involved with creating a deployment environment, such as using Setup Manager to create answer files and distribution folders for unattended Setup, using unattended Setup with Windows Server 2003 R2, and using Sysprep with disk imaging. For information on using RIS, see Chapter 28.

To perform a manual installation of Windows, go to the "Installing Windows" section of this chapter.

Using Setup Manager

Setup Manager is a Windows program that provides a wizard-driven interface for creating or modifying answer files and distribution folders. This is the fastest, easiest, and least error-prone way to create answer files and distribution folders.

Setup Manager can create the following types of answer files:

- Unattend.txt answer files for automating Windows Setup from a distribution share or a Windows CD-ROM.
- Sysprep.inf answer files for automating Mini-Setup, the abbreviated Windows Setup process that runs the next time the computer starts after using Sysprep to reseal a computer for imaging.
- A RIS .sif answer file for automating the Client Installation Wizard when installing Windows using RIS. (See Chapter 28 for more information about RIS.)

Note Original equipment manufacturers (OEMs) should use the version of Setup Manager that ships with the OEM Preinstallation Kit (OPK).

To use Setup Manager to create an answer file, and optionally a distribution folder, follow these steps:

1. Insert the CD-ROM with the most up-to-date version of the Windows Server 2003 Support Tools available, or download updated Support Tools from the Microsoft Web site. (Microsoft usually updates the Support Tools with each Service Pack. Do not use the original Windows XP Setup Manager to deploy Windows Server 2003.)
2. Navigate to the \Support\Tools folder, and extract the contents of the Deploy.cab file to a location on the hard drive of the file server on which you want to create the distribution share.
3. Launch Setupmgr.exe from the location on the hard drive to which you copied the contents of the Deploy.cab file. If you want to create a distribution share, launch Setup Manager from the server on which you want to create the distribution share.
4. On the New Or Existing Answer File page, choose Create New to create a new answer file.
5. On the Type of Setup page (shown in Figure 5-1), choose the type of setup for which you want to create an answer file.

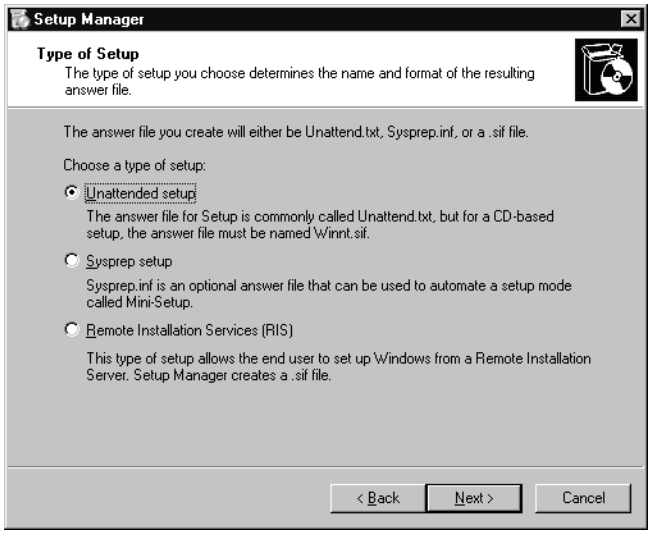

Figure 5-1 The Type Of Setup page

6. On the Product page, choose the edition of Windows for which you want to automate Setup. To automate Windows Server 2003 R2 Setup, choose the corresponding Windows Server 2003 edition.

7. On the User Interaction page, choose the level of user input you want to allow during Setup. (See Figure 5-2.) Choose Fully Automated to prevent Setup from stopping for user input. See the sidebar "Choosing an Interaction Level" for more information.

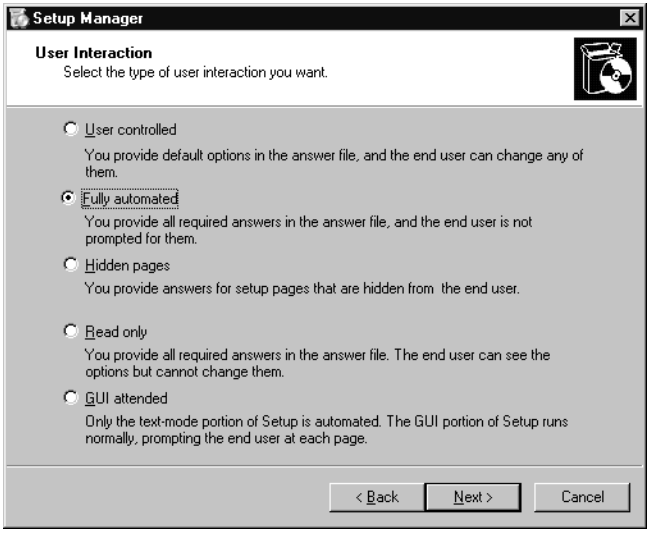

Figure 5-2 The User Interaction page

8. On the Distribution Share page, choose whether to create a distribution share, modify an existing distribution share, or create an answer file for use with a Windows CD-ROM. If you chose Set Up From A CD, skip to step 11.

   To create a distribution share for Windows Server 2003 R2, you must perform additional steps after using Setup Manager, as discussed in the "Using Unattended Setup with Windows Server 2003 R2" section of this chapter.

9. On the Location Of Setup Files page, choose On The CD to copy the installation files from a Windows CD-ROM that matches the edition of Windows you want to deploy, or choose In The Following Folder and then specify the location of the \i386 or \amd64 folder.

10. On the Distribution Share Location page, specify the location for the distribution folder and the share name.

11. On the License Agreement page that appears if you chose to fully automate Setup, select the I Accept The Terms Of The License Agreement option to agree to the terms of the End-User License Agreement (EULA) on behalf of the end-user.

12. Use the Setup Manager window (shown in Figure 5-3) to specify additional settings. You do not have to fill out every setting; however, you must use the Name And Organization, Time Zone, Product Key, and Computer Names pages when creating a fully automated answer file.

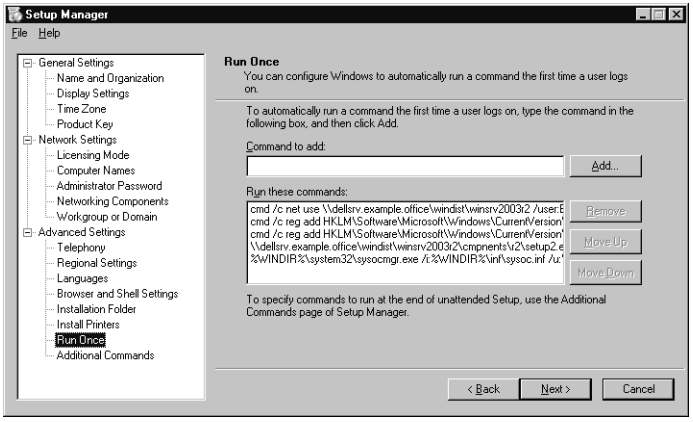

Figure 5-3 The Setup Manager window

The following pages are noteworthy:

- **Computer Names** Use this page to create or import a list of computer names to use for your systems. Setup Manager takes the names (if you have two or more) and creates a uniqueness database file (UDF) that Setup then queries for computer names, using each name only once. Use computer names that are both DNS-compatible and NetBIOS-compatible if pre–Windows 2000 clients connect to these computers over the network.
- **Administrator Password** Use this page to specify the local Administrator account passwords for the systems. If you choose to specify a password now, select the Encrypt The Administrator Password In The Answer File check box. If you do not select this check box, the password is stored in plain text for anyone with Notepad to read.
- **Workgroup or Domain** Use this page to specify the domain or workgroup the computers are to join. To join the computers to the domain, choose the Domain option and then type the Windows Server domain name in the text box provided. Windows prompts the user for a valid domain user account and password when he or she logs on to the domain after Setup completes.

Important Although Windows Setup Manager can encrypt the administrator password, it cannot encrypt the user name and password you specify for creating the computer accounts. Because of this serious security flaw, avoid using answer files to create computer accounts. Instead, create the computer accounts beforehand on the server. See Microsoft Knowledge Base Article 315273 for information about scripting the creation of computer accounts. If you feel you must use an answer file to create computer accounts, guard the file carefully until after installation is completed. (Setup deletes the user name and password from the answer file during the setup process.)

- **Run Once** Use this page to specify commands to run the first time a user logs on. You can use this page, which corresponds to the [GUIRunOnce] section of the Unattend.txt, to automate Windows Server 2003 R2 Setup (Setup2.exe), or the Active Directory Installation Wizard (as discussed in Chapter 14). You can also use this page to install applications that can be silently installed from a command prompt (and that do not require a reboot). When using this page to install applications, use the When A Destination Computer Starts, Automatically Log On As Administrator check box on the Administrator Password page to force the applications to install during the automated setup process rather than the first time an actual user logs on.
- **Additional Commands** Use this page to specify commands to run near the end of GUI-mode Setup, before the computer reboots and the OEM folder is deleted. You can use this page to run an application or command that does not require network connectivity, Windows installer support, or a locally logged-in user. Setup Manager ignores this page unless you are creating a new distribution share or modifying an existing one. This page creates a Cmdlines.txt file in the \i386\$OEM$ or \amd64\$OEM$ folder of the distribution share.

More Info For more information about the [GUIRunOnce] section and the Cmdlines.txt file, see the "Using Cmdlines.txt" topic in the Windows XP Resource Kit at *http://www.microsoft.com/resources/documentation/Windows/XP/all/reskit/en-us/Default.asp?url=/resources/documentation/Windows/XP/all/reskit/en-us/prbc_cai_jteq.asp*.

13. On the Additional Commands page, click Finish, type the name and location for the answer file, and then click OK.

    Setup Manager creates the distribution folder (if you chose to create a new distribution folder) and creates the answer file. It also creates a batch file that starts Setup (Winnt32.exe) in unattended mode using the answer file you created. You can use this batch file or launch Setup with the appropriate parameters, including the parameter pointing to the answer file. (See Tables 5-3 and 5-4 for the appropriate parameters.)

14. If you are creating an answer file for a CD-ROM–based installation, rename the Unattend.txt file **Winnt.sif**, place it on a floppy disk or USB Flash Device (UFD), and insert the disk or UFD immediately after the computer starts from the Windows CD-ROM.

15. If you plan to start Windows Setup from an MS-DOS environment such as a Windows 98 Startup Disk, modify the batch file that Setup Manager creates to use the 16-bit Setup (Winnt.exe) file by deleting the **32** from the third-to-last line of the file. Also change the */unattend:* parameter in the second-to-last line to */u:*.

Real World Choosing an Interaction Level

The level of user interaction you require determines how much the person running the installation needs to attend to the process. Here is a more detailed explanation of the interaction levels:

- **User Controlled** Uses the information in the answer file as default answers during the Windows installation. The user must confirm the defaults or make changes as the installation progresses. (Nothing is automated.)
- **Fully Automated** Completely automates Setup by using the settings you specify in the answer file. This option is best for quickly setting up multiple systems with identical configurations. If any settings do not work properly (such as might happen if the answer file provides a computer name that is already in use), the installation fails.
- **Hidden Pages** Automates the parts of Setup for which you provide information, and prompts the user to supply any information you did not include in the answer file. Use this option to set up a system in a specific way, while still allowing the user some limited customization options. (The user sees only the parts of Setup that the answer file does not cover.)
- **Read Only** Hides the pages for which you provide information, just like the Hidden Pages option. However, if the answer file does not provide information for all settings on a page, Setup displays the page and prompts the user to complete the unanswered portion of the page. You cannot change settings provided in the answer file during installation.
- **GUI Attended** Automates the text-mode portion of Setup. The person running Setup supplies answers for the Windows Setup Wizard. Use this option when you want to automate the text-mode portion of Setup and allow the person running the installation to provide the settings during the graphical user interface (GUI) portion.

Note If you create the system partition in a preinstallation environment without restarting the computer, add the */syspart* and */tempdrive* parameters to the Winnt32.exe command in the Unattend.bat batch file or the command that you use to start Setup.

Using Unattended Setup with Windows Server 2003 R2

To automate Windows Server 2003 R2 Setup using an answer file, you must add the \Cmpnents folder from Windows Server 2003 R2 Disc 2 to the distribution share or RIS image, and add commands to the answer file to launch Windows Server 2003 R2 Setup (Setup2.exe). If you do not perform these additional steps, Windows Setup stops after installing Windows Server 2003 with Service Pack 1, and prompts you to insert the second CD-ROM to continue Windows Setup.

To prepare the distribution share for Windows Server 2003 R2, copy the \Cmpnents folder from the Windows Server 2003 Disc 2 CD-ROM into a Windows Server 2003 with Service Pack 1 distribution share, at the same level as the \i386 or \amd64 folder. For example, if the path to the \i386 folder on the distribution share is \\srv2\windist\winsrv2003\i386, copy the \Cmpnents folder to \\srv2\windist\winsrv2003\Cmpnents.

Note To prepare a RIS image for Windows Server 2003 R2, copy the \Cmpnents folder into the image at the same level as the \i386 or \amd64 folder, associate a new RIS .sif answer file to the Windows Server 2003 with Service Pack 1 image, and then modify the answer as discussed in this section. See Chapter 28 for information about associating answer files to RIS images.

To launch Windows Server 2003 R2 Setup from an Unattend.txt, Sysprep.inf or RIS .sif answer file, follow these steps:

1. Open the appropriate answer file in Setup Manager and then navigate to the Run Once page.
2. If you are running Windows Setup from a distribution share, type the following command in the Command To Add box and then click Add.

   "cmd /c net use ***srv2.example.office****windist **/user:***EXAMPLE******Install Password**"**

   This command connects to the distribution share and authenticates with the server, if necessary. Replace *srv2.example.office**windist* with the UNC name of the distribution share. Replace *EXAMPLE**Install* with the domain (or computer for workgroup environments) and username of a limited-rights user account with permission to access the share, and *Password* with the password for the user account.

3. If you are creating an Unattend.txt file for installation from a distribution share or local source other than a CD-ROM or DVD-ROM, and you need to copy files from the OEM folder and subfolders during Setup, type the following commands in the Command To Add box. Click Add after each line, and replace *source_path* with the path to the location of the \i386 or \amd64 and \cmpnents folders:

   "cmd /c reg add HKLM\Software\Microsoft\Windows\CurrentVersion\Setup /v SourcePath /t REG_SZ /d *source_path* **/f"**

   "cmd /c reg add HKLM\Software\Microsoft\Windows\CurrentVersion\Setup /v ServicePackSourcePath /t REG_SZ /d *source_path* **/f"**

   These commands ensure that the SourcePath and ServicePackSourcePath registry keys are set to the location of the \Cmpnents and \i386 (or \amd64) folders when Windows Server 2003 R2 Setup runs during the [GUIRunOnce] processing and for subsequent installations of Windows Server 2003 components (including Windows Server 2003 R2 components). Otherwise, the SourcePath and ServicePackSourcePath keys are reset to the drive letter of the CD-ROM drive (usually D).

4. Type the following command in the Command To Add box, and then click Add. This command launches Windows Server 2003 R2 Setup from the [GUIRunOnce] section of the Unattend.txt, Sysprep.inf or .sif answer file.

   "*srv2.example.office**windist***cmpnents\r2\setup2.exe /q /a**"

5. If you want to automate the installation of Windows Server 2003 R2 optional components, type the following command in the Command To Add box, and then click Add.

   "%WINDIR%\system32\sysocmgr.exe /i:%WINDIR%\inf\sysoc.inf /u:*srv2.example.office**windist**Sysocmgr.ini***"**

   This command launches Sysocmgr.exe in unattended mode using the Sysocmgr.ini answer file you specify (as described later in this chapter).

6. If you are creating an Unattend.txt file for installation from a distribution share or local source and you do not need to copy files from the OEM folder and subfolders during Setup, open the Unattend.txt file in Notepad or another text editor and change **OEMPreinstall = Yes** in the [Unattended] section to **OEMPreinstall = No**.

   Doing so prevents Setup from resetting the SourcePath and ServicePackSourcePath registry keys to the drive letter of the first CD-ROM drive (usually D:\).

Note If you are using the [OEMRunOnce] section of Winbom.ini to install Windows Server 2003 R2 during Sysprep factory mode, you can use the `SourcePath` entry in Winbom.ini to set the SourcePath and ServicePackSourcePath registry keys during Sysprep factory mode.

Installing Optional Components During Unattended Setup

You can install most optional Windows components during unattended Setup by using the [Components] section of the Unattend.txt file. You cannot install Windows Server 2003 R2 components (or Message Queuing) during Windows Setup, but you can install them automatically after Setup completes by using Sysocmgr.exe and an answer file (Sysocmgr.ini). You can also use Sysocmgr.exe with an answer file to install or uninstall any Windows component at any time after Setup completes.

To install Windows components automatically using Sysocmgr.exe, create a text file named Sysocmgr.ini (or any other name) that includes a listing of the components that you want to install using the following syntax:

```
[Components]
DFSFRSUI = On
DFSR = On
Netfx20 = On
```

Then run the Sysocmgr.exe command with the Sysocmgr.ini file you created, as described in Step 5 of this section. For more information about Sysocmgr.exe and Sysocmgr.ini, see the Microsoft Windows Corporate Deployment Tools User's Guide (Deploy.chm and Ref.chm) on Windows Server 2003 Disc 2 or at the Microsoft Download Center (*http://www.microsoft.com/downloads/*).

Creating and Modifying a Distribution Share

A distribution share is a shared folder on a server from which you can install Microsoft Windows. The distribution share must contain the Windows installation files: the \i386 or \amd64 folder from the Windows CD-ROM, the \Cmpnents folder (when installing Windows Server 2003 R2), and any device drivers or other files that you want to include during Windows Setup.

This section discusses how to create a distribution share manually, and how to modify a distribution share or RIS image by adding drivers, software updates, and service packs.

Creating a Distribution Share

The easiest way to create a distribution share is to use Setup Manager, as described earlier in this chapter. To create a distribution share for RIS, use the Remote Installation Services Setup Wizard, as described in Chapter 28. However, you can also create a distribution share manually or modify the files contained within it. To do so, follow these steps:

1. Create a folder on the server named \Windist (for example), and share it on the network with the proper share name and permissions. (Keep the share name to eight characters or fewer if it needs to be accessible from an MS-DOS–based or Windows 98–based boot disk.)

   Give Administrators Full Access, and give other users Read and Execute permissions.

2. Create a descriptively named subfolder (such as W2K3), and copy the \i386 or \amd64 folder from the Windows CD-ROM into the subfolder.

3. If you are creating a distribution share for Windows Server 2003 R2, copy the \Cmpnents folder from Windows Server 2003 R2 Disc 2 into the same folder in which you copied the \i386 or \amd64 folder.

4. Create a subfolder named OEM in the \i386 or \amd64 folder (or at the same level as the \i386 or \amd64 folder for RIS installations) for any additional drivers or programs that you want to preinstall. Setup copies all files and folders in this folder into the temporary setup folder during installation, and then deletes them near the end of GUI-mode Setup.

   You can also place the \$OEM$ folder outside the distribution share if you place the path (file or UNC) to the \$OEM$ folder after the OEMFilesPath key in the answer file.

5. Create any additional subfolders needed for the installations. Table 5-4 describes the special folders that you can create for use by Setup.

Note If you are using Winnt.exe to launch Windows Setup, all files and folders that you create in the distribution share must have short (8.3) file names. To convert files with short filenames back to long filenames during Setup, see the "Converting Short Filenames Back to Long Filenames" section of this chapter.

Table 5-4 Subfolders you can create to store extra files

Folder	Description
\$OEM$\$1	The folder in which you place files that you want Setup to copy to the drive on which it installs Windows. Equivalent to the *%systemdrive%* environment variable, you can use the \$OEM$\$1 folder to change drive letters without causing problems for applications that point to a hard-coded drive letter. You can also create subfolders here for the files; Setup copies the entire folder structure to the system drive. If you use the \$OEM$\$1 folder, set OEMPreinstall = Yes in the [Unattended] section of the Unattend.txt answer file.

Table 5-4 Subfolders you can create to store extra files

Folder	Description
\$OEM$\$1\Sysprep	The folder in which you place the files required to run Sysprep, if you want to run Sysprep at the end of unattended installation to reseal the computer or install additional applications using Sysprep Factory mode.
\$OEM$\$$	The folder that holds any new system files or files that replace existing system files. Setup copies these files into the various subdirectories of the Windows system folder (\Windows). This folder must match exactly the structure of the Windows system subfolders for those folders in which you want to add or replace system files. For example, to copy new or replacement files into the *%windir%*\System32 folder, create an \$OEM$\$$\System32 folder.
\$OEM$\$Progs	The folder that holds any files that you want to copy to the %PROGRAM FILES% folder during Setup.
\OEM\textmode	The folder in which you place any hardware-dependent files for use while loading Windows Setup and during the text-mode phase of Setup. These files include updated SCSI, keyboard, video, and pointing device drivers. Include the Txtsetup.oem file in this folder to control the loading and installation of these files. To create the Txtsetup.oem file, create a normal text file and list the filenames of all files in this folder. List the Txtsetup.oem file and all files mentioned in the Unattend.txt file, under the [OEMBootFiles] section.
\OEM*drive_letter*	The folder that specifies additional files and folders that you want Setup to copy into the root folder of the named drive. Create one entry for each drive to which you want to copy files. For example, Setup copies the files located in the \$OEM$\C folder into the root folder of the C: drive during the text-mode phase of Setup. Setup also copies any subfolders of the \$OEM$\C folder.

Note Windows Server 2003, Windows XP, and Windows 2000 do not use the \$OEM$\Display and \$OEM$\Net subfolders that Windows NT 4 Setup used.

Applying Service Packs to a Distribution Share

You can apply a service pack to a distribution share or standard (non-RIPrep) RIS image so that subsequent installations of Windows are "integrated" installations of Windows that include the latest service pack. To perform this procedure, follow these steps:

1. Make a copy of the distribution share or RIS image so that you do not affect existing clients when you upgrade the distribution share.

   Existing clients that do not have the latest service pack installed might access the distribution share to install optional Windows components, and they must receive the version of the components that corresponds to the service pack level of the client.

2. If the Windows install files are not located in the \i386 or \amd64 folder, move them there before applying the service pack.
3. Open a command prompt, switch to the folder storing the service pack, and then apply the service pack to the distribution share or RIS image.

   To do this for Windows Server 2003 Service Pack 1, type **WindowsServer2003-KB889101-SP1-x86-ENU.exe /integrate:***path*, where *path* is the path to the distribution share you want to update (for example, e:\windist\winsrv03sp1). The parameters vary depending on the service pack—consult the service pack documentation for further information.

When you run the Update.exe file to update a Windows distribution share with the latest service pack, it creates a Svcpack.log file in the *%systemroot%* of the computer from which you ran Update.exe. If you want to perform more service pack installations from the same computer, rename this log file before performing them.

Installing Software Updates with an Answer File

Most organizations choose to deploy software updates after Setup completes using their chosen software update management solution. However, you might want to apply some updates during Setup. For example, you might want to deploy Windows Server 2003 R2 with a set of tested and approved security updates to maximize the security of the server from the moment Setup is complete. This is particularly important when there are viruses on your internal network that can infect new systems before you can install antivirus software and the latest security updates. See Chapter 23 for more information about software updates.

Important Install software updates released only after the service pack version you are deploying. For example, if you are deploying Windows Server 2003 R2 (which is based on Windows Server 2003 with Service Pack 1), add only hotfixes that are labeled sp2 (which means they are post–Service Pack 1).

The most flexible method of installing software updates during Setup is to use the following steps to copy the updates into the distribution share and add them to an answer file.

1. Copy the update executable file into the OEM folder.
2. On the Additional Commands page in Setup Manager, type the filename of the update followed by the quiet (**/q**) and unattended (**/u** or **/m**, depending on the update version) parameters. (This adds the commands to the Cmdlines.txt file.) For example, type **KB123456.exe /q /u**.

   Some updates use different parameters—to view the parameters supported by an update, type the filename followed by the **/?** parameter.

To suppress a reboot when installing an update that requires a reboot, add the **/z** parameter to the software update command. (Not all updates support or require this parameter.)

You can apply software updates to a distribution share in the same way as you apply a service pack: run the update with the /integrate parameter. However, you cannot uninstall software updates applied to a distribution share, so most organizations prefer to add software updates to an answer file or install the updates using the organization's chosen software update solution after Setup completes.

Note Hotfix chaining (that is, installing multiple software updates with a single restart) is supported in Windows Server 2003, Windows XP, and Windows 2000 Service Pack 3 and later. Previously, you had to restart after each software update installation, or run the Qchain.exe tool after installing the updates before restarting. Install software updates in order by date or Knowledge Base article number when possible.

Installing Plug and Play Drivers in the Distribution Share

To add Plug and Play (PnP) drivers to a distribution share or RIS image, follow these steps:

1. Create a subfolder in the \$OEM$\$1 folder with a name containing no more than eight characters. This folder remains on the destination computer's hard drive after Setup completes. If you want to protect these drivers from accidental deletion by end users, instead create it in the \$OEM$\$$ folder, which Setup copies to the \Windows folder.

   If you are adding drivers to a RIS image, create the \OEM\ folder at the same level as the \i386 or \amd64 folder.

2. Inside the subfolder you created, you can make additional subfolders to categorize the devices. For example, you might have the following directories:

   ```
   \$OEM$\$1\Drivers\Net
   \$OEM$\$1\Drivers\Video
   \$OEM$\$1\Drivers\Sound
   ```

3. Copy the drivers and .INF files into the appropriate subfolder.

   If you are installing a network driver for a RIS image, copy the driver files into the \i386 or \amd64 folder in addition to the subfolder you created. If this driver replaces an existing driver in the \i386 or \amd64 folder, delete the existing driver's .pnf file.

4. Add the appropriate lines to the answer files:

   - **Standard answer file users** Add each subfolder to the OEMPnPDriversPath entry of the [Unattended] section of the answer file (Unattend.txt or Sysprep.inf), separating each folder reference by a semicolon. For example:

     ```
     OEMPnPDriversPath = Drivers\Nic;Drivers\Video;Drivers\Sound
     ```

   - **Remote Installation Services (RIS) users** Modify the following lines or add them to the [Unattended] section of the default template for the desired image (Ristndrd.sif):

     ```
     OemPreinstall = Yes
     OEMPnPDriversPath = Drivers\Nic;Drivers\Video;Drivers\Sound
     ```

     Note Leave the drive letter out of the paths. Setup automatically adds the system drive to the paths.

     If you are installing new drivers for a RIS-based operating system image, restart the BINL (Boot Information Negotiation Layer) service on the RIS server after copying the files into the distribution folder. To do so, type the following commands at a command prompt:

     ```
     net stop "boot information negotiation layer"
     net start "boot information negotiation layer"
     ```

Note Sysprep and RIS installations postpone installing devices with unsigned drivers until an administrator logs on to the computer. To avoid this (at your own peril), add the **DriverSigningPolicy = Ignore** line to the [Unattended] section of the answer file.

Installing OEM Drivers in Remote Installation Preparation Images

Although it is tempting to add OEM drivers to a Remote Installation Preparation (RIPrep)-based image by installing them on the source computer before running the RIPrep program on it, this does not work because RIPrep images need to be able to adapt to a wide variety of hardware. Therefore, follow these steps to add drivers to RIPrep-based images:

1. Before running the RIPrep process (described in Chapter 28), create a folder named Sysprep on the C:\ drive of the source computer.
2. Create a C:\Drivers folder to hold OEM drivers, and create any necessary subfolders, such as C:\Drivers\Video or C:\Drivers\Sound.
3. Create a Sysprep.inf file, and place it in the Sysprep folder.

4. Add the following lines to the Sysprep.inf file:

```
[Unattended]
OEMPnPDriversPath = Drivers\Nic;Drivers\Video;Drivers\Sound
```

5. Use Device Manager to remove any devices for which you are installing updated or new drivers. Also, remove any unknown devices.

6. If you are adding mass storage drivers that Windows requires to start, use the "Installing Mass Storage Drivers" section of this chapter to add the drivers to the appropriate RISetup image (a flat RIS image created using Risetup.exe or the Remote Installation Services Setup Wizard). Add the drivers to the RISetup image that matches the operating system version (not including service packs) of the RIPrep image and whose folder name is listed first alphabetically.

   For example, when adding mass storage drivers to a Windows Server 2003 R2 RIPrep image on a Windows Server 2003 RIS server that contains RISetup images in the \Win2K3 and \Win2K3R2 folders, add the drivers to the RISetup image located in the \Win2K3 folder, because \Win2K3 comes before \Win2K3R2 alphabetically.

7. Run Riprep.exe from the RIS server's network share, as described in Chapter 28.

8. Stop and restart the BINL service on the RIS server by typing the following commands:

```
net stop "boot information negotiation layer"
net start "boot information negotiation layer"
```

Under the Hood Why RIPrep Images Need RISetup Images

A RIPrep image does not have a fully populated critical device database, so it uses the critical device database from a matching RISetup image until the client can load Windows and perform its own enumeration of devices. This is why you must add mass storage drivers to both the RIPrep image and the matching RISetup image.

RIS determines which RISetup image to use for a given RIPrep image at runtime by looking at the operating system version and folder name, so the image RIS selects could change if you add or rename image folders.

Installing Mass Storage Drivers

If the hard drive or installation source is attached to a device for which Windows does not have drivers, such as a new network adapter or Serial ATA (SATA) controller, you have a problem. When performing a manual installation, you can always press F6 at the beginning of text-mode setup to install new mass storage drivers from a floppy disk, but this option is cumbersome with unattended installations. To get around this, follow these steps to copy mass storage drivers that are required at boot time to the distribution share or RIS image:

1. Create a folder named \OEM\Textmode in the \i386 or \amd64 folder of the distribution share. (This step is unnecessary for a RIS image.)
2. Copy the storage drivers into the \$OEM$\Textmode folder (or the \i386 or \amd64 folder for a RIS image) as well as the folder specified in the OEMPnPDriversPath entry of the answer file (for example, \$OEM$\$1\Drivers\Storage). These files should include at least one .SYS file and the Txtsetup.oem file.
3. In the answer file, add the following lines (refer to the Txtsetup.oem file for the exact syntax):

   ```
   [MassStorageDrivers]
   "scsi controller string from txtsetup.oem" = "OEM"
   ```

   For example:

   ```
   [MassStorageDrivers]
   "DELL PERC 2/3/4 RAID Controller Driver" = "OEM"
   ```

   To enable client computers to start from a device connected to an IDE controller for which Windows provides drivers, add the following line to the [MassStorageDrivers] section:

   ```
   "IDE CD-ROM (ATAPI 1.2)/PCI IDE Controller" = "RETAIL"
   ```

4. Add or modify the [Unattended] section of the answer file to include the following line:

   ```
   OemPreinstall = Yes
   ```

5. Add or modify the [OEMBootFiles] section of the answer file to include a listing of all files in the OEM\Textmode folder. For example:

   ```
   [OEMBootFiles]
   mraid35x.cat
   nodev.inf
   Oemsetup.inf
   mraid35x.sys
   txtsetup.oem
   ```

6. Open the Txtsetup.oem file using Notepad. In the [Disks] section, remove any floppy disk references or other path references, and type a backslash (\) at the end of the line (or a period if installing on a FAT partition). For example, change the following line:

   ```
   d1 = "Windows Server 2003 Driver Set 1.01", \w23dsk1, \
   ```

7. Verify that the Txtsetup.oem file contains a section named [HardwareIds.Scsi.*device_service_name*], where *device_service_name* is the device service name. If this section does not exist, create it using the following syntax, where *device_identifier* is the device identifier:

   ```
   [HardwareIds.scsi.device_service_name]
   id = "device_identifier", "device_service_name"
   ```

   For example, for a Dell PERC RAID controller, use the following lines:

   ```
   [HardwareIds.scsi.DELLPERC]
   id = "PCI\VEN_1000&DEV_0407&SUBSYS_05311028", "mraid35x"
   ```

Converting Short Filenames Back to Long Filenames

If you start Windows Setup using the Winnt.exe setup program, or if you use MS-DOS to copy the distribution share to a local source, all files included in the distribution share must have MS-DOS–compliant short names. This is because MS-DOS and Winnt.exe will discard long filenames when copying files. However, you can convert the short filenames back to long filenames during Setup by creating a renaming file for each folder in which there are files you want to convert.

The easiest way to create a renaming file ($$Rename.txt) is to copy your files into the OEM folder (or a subfolder), open the answer file in Setup Manager that corresponds to the distribution share, and then save the answer file. Setup Manager automatically creates the necessary $$Rename.txt files for you. You do not need to rename files with long filenames because Setup Manager lists the short names of each file as MS-DOS would see them.

To create a renaming file manually, follow these steps:

1. Open Notepad, type the path to the subfolder containing the files you want to rename, and enclose the path in brackets (by leaving a blank or using the backslash character [\] for the root folder).
2. Underneath the bracketed heading, type each short filename you want to rename (not enclosing it in quotes) followed by an equals sign and then the long filename in quotes.

3. Repeat step 2 with any additional subfolders in which you have files or folders that you want to rename. A sample renaming file is shown below:

```
[media]
filenm1.txt = "Your long filename here.txt"
ding.wav = "Really loud and annoying ding.wav"
whiz.mpg = "Whizbang Deluxe Video.mpg"
[images]
desktp1.bmp = "corporate logo.bmp"
desktp2.bmp = "division logo.bmp"
```

4. Save the file as $$Rename.txt in the folder of the distribution share that contains the files that you want to convert.

Using Sysprep with Disk Imaging

One way to increase the speed of the setup process is to create a disk image of an existing reference system and then apply that image to the target system. Installing a disk image created with Sysprep and disk-imaging software usually takes 45 to 60 minutes less time than an unattended Setup-based installation.

> **More Info** For in-depth information about Sysprep, see the Microsoft Corporate Deployment Tools User's Guide (*Deploy.chm*).

Disk imaging works like this: first, install Windows and all the applications you need on a single machine that is identical or very similar to the many machines on which you want to deploy Windows. Then prepare this system for imaging by using the reseal functionality of Sysprep (which is available on the Windows CD-ROM in the Deploy.cab file) to clear out the SID and other computer identity information. Image the configuration using a disk-imaging program, such as PowerQuest Drive Image or Norton Ghost, which copies and compresses the disk image to a network share. You can then start a blank system by using Windows PE or the floppy disk created by the disk-imaging program, copy and uncompress the disk image onto the new system, and be up and running in far less time than is required to perform a fully automated Setup-based installation.

However, this solution has some problems. First, the systems must be similar for the disk images to work on them. The computers do not have to be identical because Windows uses Plug and Play (PnP) to detect changes to most system components. However, the systems must share the same HAL—no mixing ACPI systems with non-ACPI systems. If the systems use different mass storage controllers, you must add the drivers to the image by adding the drivers to the \Sysprep folder on the target system and using the [SysprepMassStorage] section of Sysprep.inf. In addition, this process does not work for Cluster service servers, Certificate Server servers, or domain controllers (unless you script the DCPROMO process into the disk image).

> **Note** Because you cannot install a Sysprep-created operating system image on a disk partition smaller than that on which you created the image, install Windows into a partition just large enough for Windows and any necessary applications. The disk-imaging software usually partitions the target computer's hard disk to the same size as the reference computer's partition, but you can use the ExtendOemPartition entry in the Sysprep.inf file to extend the destination computer's system partition to fill the hard disk.

Covering in detail how to use Sysprep and disk-imaging tools to mass deploy computers is outside the scope of this book; however, the following steps summarize how to create a disk image of a reference system and apply it to a target system:

1. Install and customize Windows on the reference system.
2. Install and customize any applications you want to deploy to *all* systems using this drive image.
3. Copy Sysprep.exe, Factory.exe, and Setupcl.exe from the Deploy.cab file that ships with the version of Windows that you are imaging into the C:\Sysprep folder.

   To automate Mini-Setup, copy a Sysprep.inf answer file you created with Setup Manager to the C:\Sysprep folder. To use Sysprep factory mode to customize an installation after applying the reference image to the target computer, create a Winbom.ini file and copy it to the C:\Sysprep folder.
4. Launch Sysprep, and click Reseal to remove all identity information from the system.
5. Restart the computer into your preinstallation environment and then use a disk-imaging program to save the disk image to the desired network share.

> **Note** After using Sysprep to reseal a reference system for imaging, you must restart the computer and run through Mini-Setup (described in step 8) before you can perform additional customizations or use it as a normal system.

6. Boot the target system with Windows PE or a network floppy disk and then connect to the network share containing the drive image.
7. Use the imaging software client tools to expand the image file onto the target system's hard drive.
8. Restart the target system. Mini-Setup runs, detects any additional PnP devices and hides any missing devices. The wizard generates a new SID, and the system is then fully functional. (To force Mini-Setup to perform a full PnP detection process to eliminate rather than hide any missing devices, run Sysprep with the /PnP switch.)

Important Thoroughly test your disk images before using them to deploy Windows in a production environment.

Installing Windows

The process of manually installing Windows or automating the installation of Windows after creating your deployment environment is straightforward. First, you prepare the system for installation. Then you install Windows, either manually, with an answer file, or via Setup command-line parameters.

Preparing the System

Before you install Windows, several physical tasks remain:

- Back up any existing data on all the drives for which the server is responsible.
- Disable any disk mirroring for the duration of the setup process.
- Disconnect any serial connections to an uninterruptible power supply (UPS). UPS equipment can interfere with the setup program's ability to detect devices connected to serial ports.
- Upgrade the system BIOS to the latest version available. If the BIOS does not meet Windows' Advanced Configuration Power Interface (ACPI) standards, set the Plug and Play (PnP) OS setting to NO in the BIOS.
- Change the boot settings in the BIOS to start the computer from the proper location (usually a CD-ROM or PXE-based network boot).
- Locate any mass storage drivers or custom hardware abstraction layer (HAL) files necessary for the system.
- If setting up a headless server without a monitor or any means of input, connect the server to the appropriate terminal concentrator. Before setting up a headless server, refer to Chapter 40 for information about Emergency Management Services (EMS).

Performing a Manual Installation of Windows

The most basic way of installing Windows is to install it manually from the Windows CD or a network share. This method is fine when doing a few installations or when learning the installation process. However, it is a tedious and slow method of deploying systems en masse; so for multiple installations, automate the process using answer files, Sysprep, or RIS.

The Phases of Setup

The Windows Setup process consists of several phases that vary depending on how you initiate the installation.

- **Preinstallation** This optional phase runs if you launch Setup from a version of the Windows operating system, Windows PE, or an MS-DOS-based boot disk. During this phase, Setup gathers information (if Setup is run from Windows or Windows PE), and then copies the files necessary to boot the computer into text-mode Setup.
- **Text-mode Setup** During this phase, you select a disk partition and Setup copies the files necessary to start into the graphical user interface (GUI)-based Windows Setup Wizard.
- **GUI-mode Setup (Windows Setup Wizard)** During this phase, Setup collects more information, installs devices, finishes copying files, processes the Cmdlines.txt file, and then deletes the OEM folder and subfolders from the target computer's hard drive.
- **First boot** This is the first boot of Windows after Setup completes, and it is when Setup processes the [GUIRunOnce] section of the Unattend.txt file, followed by the Winbom.ini file if you choose to use Sysprep Factory mode.

To perform a manual installation of Windows, start the computer from the Windows CD-ROM and then follow the instructions on the screen. Following is a list of noteworthy steps:

1. To load mass storage drivers not included with Windows—such as drivers for a Small Computer System Interface [SCSI] or RAID controller—press F6 when the computer starts in Windows Setup. To specify a different HAL manually, press F5.
2. After selecting or creating a hard disk partition, specify whether Setup should convert the partition to NTFS (because it is not an NTFS partition already), format the partition (and perform a Quick NTFS format if there is no existing data on the partition), or leave the disk alone (if there is an existing file system).

3. If you are installing Windows Server 2003 R2, type your Windows Server 2003 R2 product key in the Product Key boxes on the Your Product Key page of the Windows Setup Wizard. Typing a Windows Server 2003 R2 product key instead of a Windows Server 2003 with Service Pack 1 product key during Windows Setup eliminates the need to type a Windows Server 2003 R2 product key during Windows Server 2003 R2 Setup.

> **Note** Press Shift+F10 at any time during the Windows Setup Wizard to open a command prompt from which you can do various things, such as inspect log files or launch a screen capture program.

4. If you need to specify a static IP address, or change networking settings from the defaults during Setup, choose Custom Settings on the Networking Settings page of the Windows Setup Wizard.

   If you don't have a DHCP server and don't assign an Internet Protocol (IP) address to the computer, Windows assigns the computer an automatic private IP address in the 169.254.0.0-169.254.255.255 range with a subnet mask of 255.255.0.0. For more information, see Microsoft Knowledge Base Article 220874 at *http://support.microsoft.com/kb/q220874/*.

5. Use the Workgroup Or Computer Domain page of the Windows Setup Wizard to join a workgroup or Windows domain. To create a new domain, join a workgroup or an existing domain during Setup and create the new domain later, as discussed in Chapter 7 and Chapter 14.

6. If you are installing Windows Server 2003 R2, Setup prompts you for the Windows Server CD2 the first time an administrator logs in after Setup completes. Insert the CD, or type the location of the \Cmpnents folder. To initiate Windows Server 2003 R2 Setup later, launch Setup2.exe from the \Cmpnents\R2 folder of the installation source.

7. After Setup completes on systems running Windows Server 2003 R2 or Windows Server 2003 with Service Pack 1, the Windows Server Post-Setup Security Updates window appears. Use this window to install the latest software updates and configure Automatic Updates before allowing inbound connections. For more information, see Chapter 7.

Initiating Windows Setup Using an Answer File

To perform an unattended installation of Windows from a distribution share, follow these steps:

1. Start the computer using the preinstallation environment you chose (an MS-DOS boot disk, Windows operating system, or Windows PE).
2. Create an installation partition if there is not one already, mark it active, assign it a drive letter, and format it using the NTFS file system. For example, when using Windows PE or Windows as your preinstallation environment, use the following diskpart.exe commands to erase all data on the first physical drive (and note that you don't do this on the drive from which you are running Windows), and create a single partition of the maximum size:

   ```
   Select disk 0
   Clean
   Create partition primary
   Select partition 1
   Active
   Assign letter=c
   Exit
   Format c: /fs:ntfs /q
   ```

   When using an MS-DOS-based preinstallation environment, you must format the partition using the FAT32 or FAT16 file system. However, you can convert the partition to NTFS and extend the size during Setup. (For more information, see the Microsoft Windows Corporate Deployment Tools User's Guide.)
3. Connect to the distribution share, and then launch Windows Setup (Winnt32.exe or Winnt.exe) using the batch file created by Setup Manager or via Setup command-line parameters (as discussed in the next section).

Note If you create the system partition in a preinstallation environment without restarting the computer, add the */syspart* and */tempdrive* parameters to the Winnt32.exe command in the Unattend.bat batch file or the command that you use to start Setup.

To perform an automated installation of Windows from the Windows CD-ROM, follow these steps:

1. Start the computer from the Windows CD-ROM.
2. Immediately insert the floppy disk that contains the Unattend.txt file (renamed to Winnt.sif).

   If the BIOS boot order lists the CD-ROM before the floppy disk drive, you can insert the floppy disk before starting the computer.

Note Windows Setup does not support loading mass storage drivers or answer files from USB Flash Devices (such as USB sticks or thumb-drives). If the system BIOS supports floppy emulation, you might be able to use USB Flash Devices to load mass storage drivers or answer files, though the most reliable method is to use a real floppy disk.

Initiating Setup Using Command-Line Parameters

You can streamline the setup process on a single machine by launching Windows Setup using command-line parameters; you can also use command-line parameters to specify an answer file to automate Setup completely.

To use a command-line parameter on a computer with Windows, boot the computer in Windows and open a command prompt window. Then type [***path***]**\winnt32.exe**[***parameter***], substituting *[path]* with the location of the Windows setup files, and replacing *[parameter]* with the appropriate parameter or parameters you want to use. Table 5-5 shows the available parameters for Winnt32.exe, the 32-bit version of Setup.

Note If you have access to Windows PE, you can boot from a CD-ROM to a streamlined version of Windows XP or Windows Server 2003, which you can then use to prepare the computer and launch Setup. Original equipment manufacturers (OEMs) can create customized Windows PE CD-ROMs using the OEM Preinstallation Kit (OPK); enterprise users can use the Windows PE For Corporations Toolkit, available from your account manager.

Table 5-5 Parameters for the Winnt32.exe command

Parameter	Function
/checkupgradeonly	Runs a compatibility test on the computer to see whether it has any problems that might interfere with upgrading the operating system. It saves a Winnt32.log report in the installation folder for Windows NT upgrades, or an Upgrade.txt report in the Windows folder for Windows 98/Windows Me upgrades.
/cmd:[command]	Runs the command following the */cmd:* parameter after the Windows Setup Wizard completes.
/cmdcons	Enables the use of the Recovery Mode Console at boot time for repairing failed installations. You can use this parameter only after installing Windows.
/copydir:[folder name]	Names an additional folder you want Setup to copy into the folder in which it installs Windows (\Windows for Windows Server 2003 and Windows XP, and \WINNT for Windows 2000 and Windows NT). The folder remains after Setup completes, and you can copy additional folders by using the parameter multiple times. The folder might contain drivers or other files needed after setup. For example, create an *extra_drivers* folder in the \i386 or \amd64 source folder and use the */copydir:i386\extra_drivers* parameter.

Table 5-5 Parameters for the Winnt32.exe command

Parameter	Function
/copysource:[folder name]	Names an additional folder you want Setup to copy into the folder in which you install Windows. Setup deletes the folder near the end of GUI-mode Setup.
/debug[level:filename]	Creates a debug log file with the specified level. The default creates a log file named C:\Windows\Winnt32.log with the level set to 2 (Warning).
/dudisable=yes	Disables dynamic update during setup, even if an answer file specifies dynamic update locations.
/duprepare:[folder name]	Extracts any Dynamic Update packages downloaded from the Windows Update Web site that reside in the specified folder, and prepares it for use as a local Dynamic Updates source for clients.
/dushare:[folder name]	Specifies the location for Setup to search for Dynamic Update files. To create such a share, download and extract Dynamic Update packages from the Windows Update Catalog and then run the */duprepare* command on the folder.
/emsbaudrate:[baudrate]	Specifies the baud rate to use with the EMS serial port. Valid rates are 9600 (default), 19200, 57600, and 115200.
/emsport:[comport]	Specifies the communications port (COM port) EMS must use for remote troubleshooting. (See Chapter 40 for more information.) Replace *comport* with *com1*, *com2*, *off*, or *usebiossettings*. (Com1 and Com2 work only with x86 systems.)
/m:[folder name]	Specifies the location of a folder containing system file replacements. Setup checks this folder first for files to copy and then checks the installation folder.
/makelocalsource	Tells Setup to copy all installation files to the local hard disk so that the files are available later during the installation if the Windows CD-ROM or network share is inaccessible.
/noreboot	Tells Setup not to restart after the initial Windows file copy phase of Setup is complete. This allows you to run additional commands before continuing.
/s:[sourcepath]	Specifies the location of the Windows Setup files. (The default is the current folder.) This must be a full path—for example, X:\path or \\server\share\path. To specify multiple paths for Setup to search for needed files, use multiple */s:* parameters. (You can speed transfers by specifying the path to multiple servers that host the same source files.) Setup fails if the first server is not available.

Table 5-5 Parameters for the Winnt32.exe command

Parameter	Function
/syspart:[drive letter]	Specifies the hard disk to which you want to copy the Setup startup files. Setup makes this disk drive active and then stops, allowing you to remove the disk and insert it in another computer if you want. When you boot the computer next, text-mode Setup automatically starts. You must use the */tempdrive* parameter with the */syspart* parameter (both pointing to the same drive). You can't run this command from within Windows 98 or Windows Me.
/tempdrive:[drive letter]	Specifies the drive on which you want to store temporary files during Setup. For clean installations, this also specifies on which drive to install Windows.
/udf:[id,UDF file]	Specifies the uniqueness database file (UDF) Setup uses to modify an answer file. The ID identifies data in the UDF file that Setup uses in a corresponding section of the answer file. For example, /udf:ComputerName,our_company.udf takes the Computer Name from the UDF instead of from the answer file. If you do not specify a UDF, you are prompted to insert a disk that contains the $Unique$.udf file.
/unattend	Upgrades the previous version of Windows in unattended mode, taking all settings from the previous installation. OEMs should not use this option on computers sold to end users.
/unattend:[num:answer file]	Launches Setup in unattended mode by using the answer file you provide. The *num* parameter specifies the number of seconds to wait after copying files before restarting the computer.

As you can see, many of these parameters piggyback onto other parameters, and pretty soon you can find yourself typing (and sometimes retyping) long strings at the command prompt. If you end up doing this a lot, create a batch file (a text file with the .bat extension) containing the setup command and parameters. Then simply launch the batch file instead of typing all the parameters.

To use a command-line parameter on a computer without an existing copy of Windows, boot the computer with a Windows 98 boot disk (or use Windows PE and Winnt32.exe). Then, at the command prompt, type [***[path]*****winnt.exe**[***parameter***], substituting the location of the Windows Setup files for [*path*]. Table 5-6 shows the available parameters for use only with Winnt.exe, the 16-bit version of Setup.

Table 5-6 Parameters for the Winnt.exe command

Parameter	Function
/a	Enables Accessibility functionality during Windows Setup.
/e:*[command]*	Runs the command following the */e:* parameter after the Windows Setup Wizard completes.
/r:*[folder]*	Names an additional folder you want Setup to copy into the folder in which it installs Windows. The folder remains after Setup completes, and you can copy additional folders by using multiple */r:* parameters.
/rx:*[folder]*	Names an additional folder you want Setup to copy into the folder in which you install Windows. Setup deletes this folder after Setup completes.
/s:*[sourcepath]*	Specifies the location of the Windows Setup files. (The default is the current folder.) This must be a full path, such as X:\path or \\server\share\path. To specify multiple paths for Setup to search for needed files, use multiple */s:* parameters.
/t:*[drive letter]*	Specifies the drive on which you want to install Windows and store temporary files during Setup.
/u:*[answer file]*	Launches Setup in unattended mode using the answer file you provide. You must use the */s:* switch to specify the location of the answer file.
/udf:*[id,UDF file]*	Specifies a UDF you want Setup to use to modify an answer file. The ID identifies data in the UDF file for it to use in a corresponding section of the answer file. For example, / udf:ComputerName,our_company.udf takes the Computer Name from the UDF instead of from the answer file. If you do not specify a UDF, you are prompted to insert a disk that contains the $Unique$.udf file.

Troubleshooting Installations

Installing Windows is a relatively painless process; however, when Setup fails for some reason or another, life gets more difficult. Fortunately, you can easily solve most installation problems. The follow sections cover the most common problems; you can find additional troubleshooting procedures in Chapter 40.

More Info You can find additional troubleshooting help either in the Windows Help System's troubleshooters (which is, admittedly, not much good unless you have access to a functioning Windows 2000, Windows XP, or Windows Server 2003 machine) or in the Microsoft Knowledge Base, available at *http://support.microsoft.com*.

Setup Freezes or Locks Up

Sometimes Windows Setup inexplicably locks up during the installation process. If you receive a Stop Error message, write it down and consult either the Stop Errors troubleshooter in Windows Help or Microsoft technical support.

In general, these failures are intermittent and do not come with anything as helpful as an error message. First, restart the system by pressing Ctrl+Alt+Del. Do this repeatedly, if necessary. If you get no response, press the Reset button on the computer or turn the system off, wait 10 seconds, and then turn it back on. If you see a Boot menu, choose the Windows Setup option to allow Windows Setup to attempt to continue with its installation. If no Boot menu appears, launch Setup again. In either case, do not choose to repair the installation, but instead choose to continue with Setup.

Setup usually detects that an error occurred with its last attempt to install Windows and compensates by using a safer method of installation. If Setup hangs or stops responding again, repeat this process. Sometimes, Setup hangs multiple times before it finishes installing Windows, so be persistent. If installation freezes at a particular part of Setup, try choosing simpler setup options, if applicable.

Other procedures you can use to fix setup problems are as follows:

- Disable the system cache (processor cache) in the BIOS, and then run Setup again. Consult the hardware documentation for information about the correct procedure to do this. After Setup is complete, enable the cache again to avoid a significant performance loss.
- Try adding a wait state to the RAM in the system BIOS. This can help with partially faulty RAM chips. (However, if this server is important, plan on replacing that iffy RAM before doing any critical work on the machine.)
- Verify that the same company manufactures the RAM modules and are of the same speed and type. Although this is not a necessity, it can often eliminate problems.
- Switch the order of the RAM modules, or remove some modules and try installing them again.
- Test the RAM modules for faulty RAM chips with a third-party software program. Replace any faulty modules and run Setup again.
- Check the computer for a Master Boot Record (MBR) virus by booting it from a floppy disk that you have checked for viruses, and then run a virus-checking program and scan the drives for any viruses. If you find any viruses, clean them from the system and run Setup again.

Real World ACPI BIOS Compatibility Problems

If Setup consistently freezes during the Windows-based Setup Wizard and the system has an ACPI-compatible BIOS dated January 1, 2000 or earlier, the BIOS might not function in ACPI mode with Windows. The freezes can happen at any time during the Setup Wizard, although they most frequently happen during the device-detection phase. If you suspect the BIOS is not working properly with Windows, download the latest version from the system vendor.

If you still have trouble or if no updated BIOS is available, try disabling ACPI during Setup by pressing F5 at the beginning of the text-mode phase of Setup, right after it prompts you to press F6 to install third-party storage drivers. If this does not solve the setup problems, you do not have a problem with the ACPI support in the BIOS. (ACPI support can be added back only by reinstalling Windows, usually by performing a same-version upgrade.)

You can also manually enable or disable ACPI support after the file copy phase of Setup completes, right before the computer restarts in the Windows Setup Wizard. (Sometimes you can do this after the system freezes during the Setup Wizard.) To force Windows to enable or disable ACPI support, follow these steps:

After the text-mode phase of Setup completes but before Windows restarts in the Setup Wizard, go to a command prompt and follow these steps:

1. Type attrib −r −s −h c:\txtsetup.sif at the command prompt.
2. Open the c:\Txtsetup.sif file by using the edit command or another text editor, and search for "ACPIEnable=."
3. To force ACPI support to be enabled, which sometimes fixes setup problems, change the ACPIEnable= value to 1.
4. To disable ACPI support, change the ACPIEnable= value to 0.
5. Save the file, and restart in the Windows Setup Wizard.

Again, if any steps you take reveal questionable hardware, replace the hardware before you rely on the computer to store important data or provide critical functions to users.

Setup Stops During File Copying

If Setup locks up while copying files, you might have a problem with Integrated Device Electronics (IDE) drive configuration. Try one of the following solutions.

Reboot the machine by using Ctrl+Alt+Del or Reset, and go into the system BIOS. Verify that the IDE controllers are enabled and configured properly. Make sure that the BIOS

detects any IDE hard disks or CD-ROMs properly. (You might have to restart the system and watch the display to verify this because often the BIOS does not display drives inside the BIOS.) After doing this, one or more of the following tasks might be useful:

- Check the physical jumper settings on the drives to make sure they are properly configured to have one master and a maximum of one slave per IDE channel.
- If the CD-ROM drive is on the same channel as the hard disk, move it to the secondary channel and configure it to be the master.
- Try lowering the data transfer rate for the drives; for example, configure the drives to use PIO mode 2 instead of Ultra DMA mode or Ultra 66 transfer mode.
- Check to make sure that the drives are cabled correctly and that the cables are not faulty.
- Check the hardware settings to make sure the hard disk controller is not conflicting with another device. Try removing all cards from the computer except for the display card and SCSI adapter (if you are using a SCSI drive), and run Setup again. If Setup succeeds, add the cards one by one after installation, and use the Hardware Wizard in Windows 2000 to configure the devices and troubleshoot any hardware conflicts you encounter.

Note Windows Server 2003 provides a variety of tools you can use to boot a system that does not want to start, including the Safe Mode and Last Known Good Boot options, as well as the Recovery Mode Console, which allows you command-line access to an NTFS or FAT drive that will not boot. (See Chapter 40 for more information.)

If none of this helps, try the recommendations in the previous section or consult the Microsoft Knowledge Base.

Previous Operating System Will Not Boot

When you install Windows on a computer that's already using an operating system and you choose not to upgrade, Setup creates a dual boot so that you can select which operating system you want to use at boot time.

If the computer never displays the Windows Loader menu that allows you to choose the previous operating system, the problem is most likely one of two issues: either the Boot.ini file has a timeout set to 0 (and thus doesn't display the Boot menu), or the MBR was overwritten during Setup, preventing you from starting the previous operating system even if you have the proper entry in the Boot.ini file.

Changing the Default Operating System and Boot Times

To change which operating system Windows boots by default, as well as to control how long Windows displays a choice of operating systems at boot time, follow these steps:

1. Click Start, choose Control Panel, and then choose System.
2. Click the Advanced tab, and then click the Settings button in the Startup And Recovery section.
3. Select the operating system you want to boot by default from the Default Operating System list.
4. Select the Time To Display List Of Operating Systems check box, and specify the number of seconds you want the Boot menu displayed.

You can also do this by manually changing the *timeout* value in the Boot.ini file to a value higher than 0. To do this, click the Edit button in the System Startup section of the Startup And Recovery dialog box (described in step 4), or to edit the file from a command prompt, type **bootcfg /timeout** *N*, where *N* is the number of seconds you want the Boot menu displayed. (For more information about the Boot.ini file, see Chapter 40.)

Note You can force the Windows Boot menu to display at startup by holding down the spacebar after the BIOS displays the power-on self test (POST) screens. This displays the Hardware Profile/Configuration Recovery screen. Press F3 to display the Windows Loader screen with no timeout value.

Restoring the MBR of the Previous Operating System

If the previous operating system still does not boot properly, you might need to re-create the MBR for the operating system that you previously installed. This is risky business, so make sure you have the time to reinstall your operating system and restore a backup if you run into trouble.

Note If your previous operating system is Windows NT, Windows 2000, or Windows XP, see the startup troubleshooting section of Chapter 40.

To re-create the MBR for a version of Windows 95, Windows 98, or Windows Me, use the following steps:

1. To restore a version of Windows 95, Windows 98, or Windows Me, boot your computer with a boot disk for the operating system you are unable to boot. (Verify that the disk contains the Sys.com file.)
2. Type **A:\sys c:** at the command prompt to transfer the system files from the floppy disk to the hard disk.

3. Remove your floppy disk and restart your computer. Verify that the operating system you wanted to repair boots properly before performing the next step.
4. Boot using the Windows CD-ROM for your current version of Windows.
5. When Windows Setup launches, press Enter to begin, and then press R to display the Windows Recovery Console.
6. Choose the Windows installation to log on to and then type the administrator password.
7. Type **fixmbr** to write a valid MBR for your system.
8. Type **bootcfg /list** to display which operating systems the Boot.ini file lists. If the Boot.ini does not list an operating system, type **bootcfg /rebuild** to add operating systems back to the list.
9. Restart your computer, and choose the appropriate operating system from the Windows Loader menu.
10. If the Windows Loader menu still is not available or your current version of Windows still does not boot, return to the Recovery Console (covered in steps 4 through 6) and type **fixboot c:** (where *c:* is the system drive for your current version of Windows). Restart your system, and boot it in the desired operating system.

Summary

Deploying Windows to clients and servers is an essential task on most networks. You can save time by choosing an installation method that provides the best compromise between speed, control, and complexity. You can perform small deployments quickly and easily using the Windows CD-ROM and an answer file you create using Setup Manager. For larger deployments, you can create a deployment environment that allows you to automate virtually the entire installation process by using distribution shares, RIS, ADS, SMS, or disk imaging.

Chapter 19

Using Clusters

Microsoft Windows Server 2003 supports two high availability clustering technologies: Network Load Balancing (NLB) clusters and server clusters. Microsoft does not support combining NLB clustering with server clustering. This chapter describes the two types of clustering supported by Windows Server 2003, their place in the enterprise, and their configuration and requirements. Finally, we'll take a brief look at a new Microsoft clustering technology that is designed to support high-performance computing (HPC)—Microsoft Compute Cluster Server 2003.

What Is a Cluster?

A *cluster* is a group of two or more computers functioning together to provide a common set of applications or services with a single apparent identity to clients. The computers are physically connected by hardware in the form of either a network or shared storage. The clustering software provides a common interface externally while managing the resources and load internally.

Windows Clustering provides the following benefits:

- **High availability** When a clustered application or service fails or a computer in the cluster fails, the cluster responds by restarting the application or service on another member of the cluster or by distributing the load from the failed server to the rest of the cluster.
- **Scalability** For cluster-aware applications, adding more machines to the cluster adds capabilities.
- **Manageability** Administrators can move applications, services, and data from computer to computer within the cluster, allowing them to manually balance loads and to offload machines scheduled for maintenance.

Network Load Balancing Clusters

NLB—known as the Windows Load Balancing Service in Microsoft Windows NT 4—gives TCP/IP-based services and applications high availability and scalability by combining up to 32 servers running Windows Server 2003 in a single cluster. By combining NLB with round-robin DNS, NLB clustering can scale well beyond 32 servers. Client requests for applications and services provided by the cluster are distributed across the available servers in the cluster in a way that is transparent to the client. NLB clusters are supported in all versions of Windows Server 2003.

If a server fails or is taken offline, the cluster is automatically reconfigured and the client connections are redistributed across the remaining servers. If additional servers are added to the cluster, they are automatically recognized and the load is reconfigured and distributed.

Server Clusters

Server clusters distribute the workload among the servers in a cluster, with each server running its own workload. Like other types of clusters, server clusters are scalable and highly available. In the event of a failure, applications and services that can be restarted, such as print queues and file services, are restarted transparently. Ownership of shared resources passes to the remaining servers. When the failed server becomes available again, the workload is automatically rebalanced.

Windows Server 2003 supports server clusters only in the Enterprise and Datacenter Editions. There are three basic types of server clusters supported by Windows Server 2003: single node clusters, single quorum device clusters, and majority node set clusters, as shown in Figure 19-1.

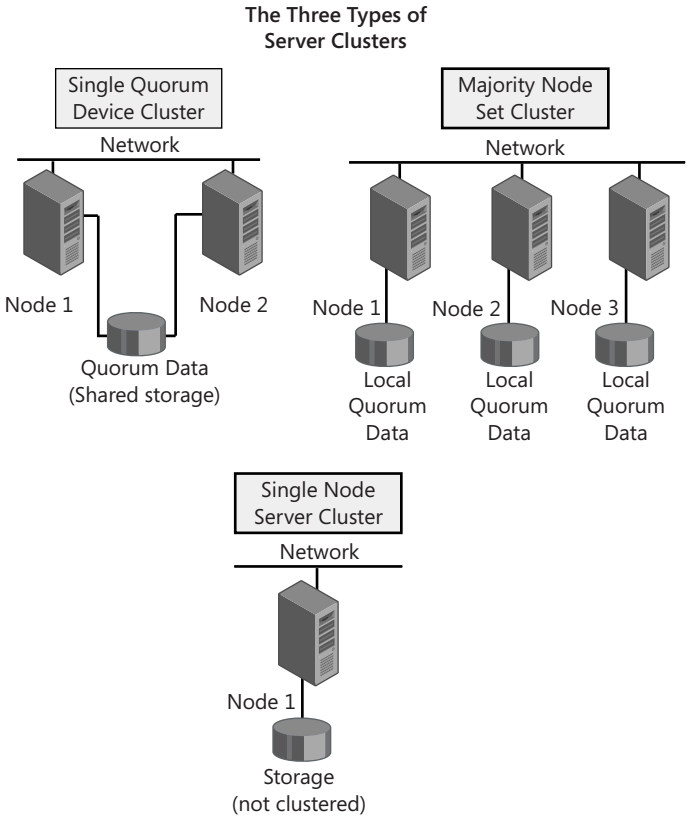

Figure 19-1 Windows Server 2003 supports three different server cluster models

Cluster Scenarios

In deciding whether and how to implement clustering, you first need to understand what problem is being solved and how best to solve it using the available technologies. Then you can make a business case for the particular solution or combination of solutions that best solves the particular problem. This section describes various scenarios and the type of clustering appropriate for each.

Intranet or Internet Functionality

An intranet or Internet server is a prime candidate for an NLB cluster. By enabling an NLB cluster across multiple servers, you provide your site with both redundancy and increased capacity. If a server should fail, the load is distributed transparently among the remaining servers.

Each Web server in the cluster runs its own Web server and accesses only local Web pages. This version is "shared nothing" clustering—there are no shared disks and no shared applications or data, with the possible exception of a common back-end database. NLB clusters are an appropriate and relatively inexpensive way to achieve both redundancy and high availability for your Web site, whether it's internal or external. Clients that need access to the Web pages are distributed among the servers in the cluster according to the load at each server. What makes this work is that most Web pages change infrequently, allowing manual updates of all Web servers with the same information when you need to make changes.

Terminal Services

Starting with Windows Server 2003, Terminal Services now supports clustering using NLB clusters and the new Session Directory to distribute Terminal Services sessions across a farm of servers running Terminal Services, allowing for high availability and load balancing and presenting a single face to Remote Desktop clients. If you have large numbers of Terminal Services users, moving to Windows Server 2003 and enabling NLB clustering for your servers running Terminal Services gives you additional flexibility, redundancy, and improved user experience for your Terminal Services users. For more information about Terminal Services Session Directory, see *http://www.microsoft.com/windowsserver2003/techinfo/overview/sessiondirectory.mspx*.

Mission-Critical Availability

If your business absolutely, positively can't be run without a certain application or set of applications, you need a highly reliable server to make sure that the application is always available. A server cluster is a good solution in this scenario, providing both high availability and scalability. With a server cluster, you organize your critical applications into groups, one group to a server. All the resources for each group are self-contained on the server, but if any server in the cluster fails, the others pick up the services and applications from the failed server, allowing for continuous availability of critical services and applications. You can control the failover and fallback actions for each server and clustered resource.

Server clusters require a substantially greater investment in hardware than NLB clusters. In addition, with the exception of majority node set clustering, they aren't suitable for "shared nothing" clustering, because they use a shared disk array to keep resources in sync. When a server fails, the other server picks up the applications that had been running on the failed server. Because the disks are shared, the remaining server has access to the same set of data as the failed server, and thus there is no loss of functionality. The exception to this is majority node set (MNS) clustering, which does not use a shared disk quorum resource but rather replicates data across the cluster to local quorum disks. Majority node set clustering is appropriate for geographically diverse clusters and requires specialized support from original equipment manufacturers (OEMs) and

independent software vendors (ISVs). For a TechNet support webcast on MNS clustering, see *http://support.microsoft.com/kb/838612*.

Requirements and Planning

Before you attempt to implement any form of clustering, you need to clearly understand the business reason for doing so. You also need to be aware of the costs and benefits of the implementation, as well as the resource requirements for a successful implementation. Treat the implementation of a Windows Server 2003 cluster as you would any other major project. Clearly state the business case for the cluster, and obtain a commitment from all levels before you expend substantial resources on the project.

Identifying and Addressing Goals

The first step in planning your cluster is to identify your goals for the implementation and the needs that using clusters will meet. This sounds obvious, but it is actually the part of the process that is most often overlooked. The implementation of any technology should always be first and foremost a business decision, not a technology decision. Creating and maintaining clusters is not a trivial task, and it requires both technological and financial resources. You'll have a hard time selling your project if you haven't clearly identified one or more needs that it will meet.

In identifying the needs to be met and the goals of your project, you need to be as objective as possible. Always keep in mind that what you might view as "cool" technology can look remarkably like scary, unproven gobbledygook to those in the organization who are less technically savvy than you are. This doesn't mean that those individuals won't support your project, but it does mean that you need to make the case for the project on a level that they can understand and identify with.

Start by clearly identifying the business goals that you're trying to accomplish. State the general goals, but provide enough detail to make the success of the project clearly measurable. Identify the specific gains you expect and how those gains will be measured. Be sure to clearly indicate how the needs you've identified are currently being met. This step is critical because it lets you point out both the costs of your suggested method and the risks associated with it.

Identifying a Solution

Once you know the business needs you're trying to meet, you can identify some solutions. If you've clearly laid out your goals and objectives for the project, the technology that achieves those goals will be driven by those needs, not the other way around. This is also the time to use your best political judgment. You need to identify not only the best

way to meet the business needs, but also how much you can realistically sell and implement in a single shot. If you think that ultimately you will need a fully integrated, three-tiered, multiple-cluster solution, you might want to build your plan around a phased approach that allows you to distribute the risks and costs over a broader period.

In addition, if you're proposing a clustering solution to the problem, spend some time and energy identifying methodologies that might be considered alternatives to clustering and clearly laying out the strengths and weaknesses of those alternatives. This effort will short-circuit objections and diversions as you build support for your project.

Identifying and Addressing Risks

As you plan your schedule, be sure to identify the risks at each step of the process and plan solid fallback positions if problems arise. Selling the project is also much easier if it's clear that you've actually thought about the risks. For example, if your goal is to replace an existing manual methodology, have you left yourself a way to fall back to it if there are problems? Or are the two mutually incompatible? If you're replacing an existing client/server application with a clustered, Web-based, distributed, *n*-tiered application, have you drawn a clear roadmap for how you will make the transition from one to the other? What are the risks of that transition?

Spend some time identifying failure points in your project. If you're building a server cluster to provide 24-hour, 7-day access to your Microsoft Exchange messaging, have you identified redundant network connections to the cluster? It does little good to create a highly available server if the network connection to it is questionable.

Making Checklists

Take the time to identify all the possible pieces of your cluster implementation ahead of time. Use this to build a checklist of steps that you need to take and the dependencies at each point. At each major step, identify the hardware, software, knowledge, and resources required, and create a checklist of the prerequisites for that step. Use the checklists in the Windows Help for Cluster Administrator as a starting point, but build onto them with the details for your implementation and your environment. The time you spend planning your clustering implementation will easily be saved in the actual installation and implementation, and it greatly reduces your risks of failure.

Network Load Balancing Clusters

NLB provides a highly available and scalable solution for TCP/IP-based network applications such as a Web server or FTP server. By combining the resources of two or more servers into a single cluster, NLB can provide for redundancy of information and resources while servicing far more clients than a single server alone could handle.

NLB Concepts

NLB is a Windows Server 2003 networking driver. It acts independently of the TCP/IP networking stack and is transparent to that stack. The NLB driver (Wlbs.sys) sits between the TCP/IP stack and the network card drivers, with the Windows Load Balancing Service (Wlbs.exe)—the necessary NLB control program—running on top, alongside the actual server application. (See Figure 19-2.)

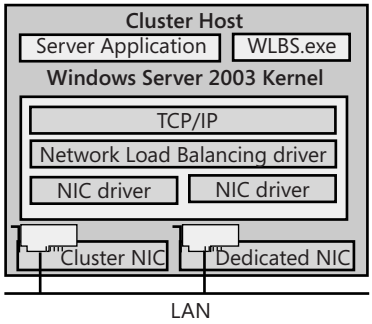

Figure 19-2 NLB as a network driver

> **Note** The Windows Load Balancing Service (Wlbs.exe) has been renamed in Windows Server 2003 to the Network Load Balancing Service (Nlb.exe). However, Wlbs.exe can continue to be used interchangeably with Nlbs.exe to provide full compatibility with existing scripts and applications. New scripts and applications should reference Nlb.exe to avoid future deprecation of Wlbs.exe.

Optimally, each server participating in an NLB cluster should have two network interface cards (NICs), although this is not an absolute requirement. Communications and management are materially improved with two NICs, however, especially in unicast mode. (Unicast mode, as opposed to multicast mode, allows each NIC to present only a single address to the network.) Overall network throughput is also improved, as the second network adapter is used to handle host-to-host traffic within the cluster. NLB clustering is not a place to try to cut costs on network cards. Server-grade NICs will provide full network throughput while minimizing the load on the servers.

NLB supports up to 32 computers per cluster. Each server application can be balanced across the entire cluster or can be primarily hosted by a single computer in the cluster, with another computer in the cluster providing directed failover redundancy. For fully distributed applications, the failure of any single host causes the load currently being serviced by that host to be transferred to the remaining hosts. When the failed server comes back online, the load among the other hosts is redistributed to include the restored server. While NLB clustering does not provide the failover protection appropriate for databases, it does provide for high availability and scalability of TCP/IP-based applications.

Note NLB is supported across the Windows Server 2003 family and requires that TCP/IP be installed. It works over Fiber Distributed Data Interface–based or Ethernet-based networks (including Wireless) from 10 megabits per second (Mbps) to 1 gigabit per second (Gbps). It uses from 250 KB to 4 MB of RAM and roughly 1 MB of disk space.

Choosing an NLB Cluster Model

A host in an NLB cluster can use one of four models, each with its own merits and drawbacks. These models are as follows:

- Single network adapter in unicast mode
- Single network adapter in multicast mode
- Multiple network adapters in unicast mode
- Multiple network adapters in multicast mode

The choice of model for a given host and cluster varies depending on the circumstances, requirements, and limitations imposed on the design of the cluster. The sections that follow provide details on each of the models.

Note NLB in Windows Server 2003 does not support a mixed unicast mode and multicast mode environment. All hosts in the cluster must be either multicast or unicast. Some hosts, however, can have a single adapter, whereas others have multiple adapters. In addition, NetBIOS cannot be supported in a single-adapter-only configuration.

Single Network Adapter in Unicast Mode

A single network adapter running in unicast mode is in some ways the easiest type of host to set up, and with only a single adapter, it is cheaper than one with multiple network adapters. It does, however, impose significant limitations:

- Overall network performance is reduced.
- Ordinary communications among cluster hosts are disabled.
- NetBIOS support is not available within the cluster.

Single Network Adapter in Multicast Mode

Using multicast mode in clusters in which one or more hosts have a single network adapter means that normal communications are possible between hosts within the

cluster. This capability overcomes one of the most awkward limitations of the single adapter in unicast mode. However, there are still the following significant disadvantages:

- Overall network performance is reduced.
- Some routers do not support multicast media access control (MAC) addresses.
- NetBIOS support is not available within the cluster.

Multiple Network Adapters in Unicast Mode

Using multiple network adapters in unicast mode is generally the preferred configuration. It does impose the cost of a second network adapter per host, but given the relatively low cost of network adapters, including the per-port cost of hubs, this is a relatively minor price to pay for the resulting advantages:

- No limitations are imposed on ordinary network communications among cluster hosts.
- Ordinary NetBIOS support is available through the first configured adapter.
- No bottlenecks occur as a result of a single network adapter.
- The model works with all routers.

Multiple Network Adapters in Multicast Mode

If you are forced by circumstances to use some hosts within a cluster that have only a single network adapter and you must be able to maintain normal network communications among the hosts in the cluster, you must run all the hosts in multicast mode, even those with multiple adapters, because you can't run some hosts in unicast mode and some in multicast mode. This limitation could cause a problem with some routers, but otherwise it is a viable solution.

Creating an NLB Cluster

Creating an NLB cluster requires using the Network Load Balancing Manager, shown in Figure 19-3. This new manager simplifies the creation and management of NLB clusters, bringing all the pieces into a single management interface. You can connect to the NLB with the NLB Manager on any address in the cluster, including private addresses or the shared public address.

New NLB Cluster

To create a new NLB cluster, follow these steps:

1. Open the Network Load Balancing Manager from the Administrative Tools folder, as shown in Figure 19-3.

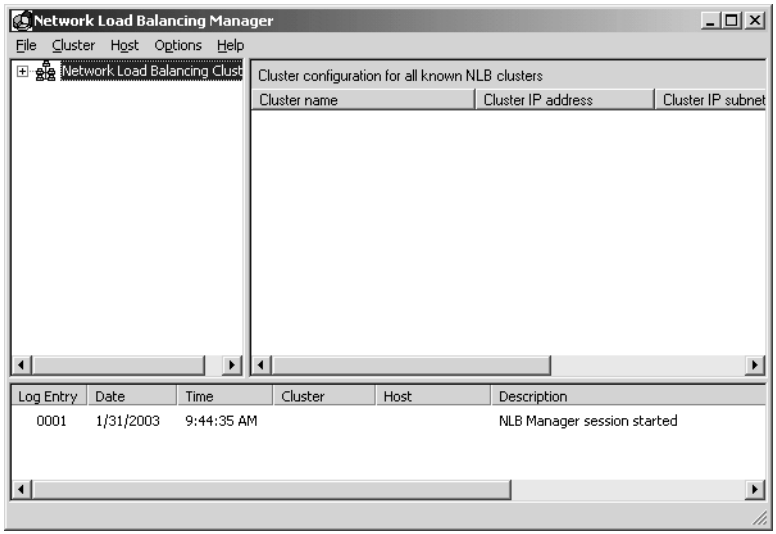

Figure 19-3 The Network Load Balancing Manager main screen

2. Right-click Network Load Balancing Clusters in the left pane and select New Cluster, as shown in Figure 19-4.

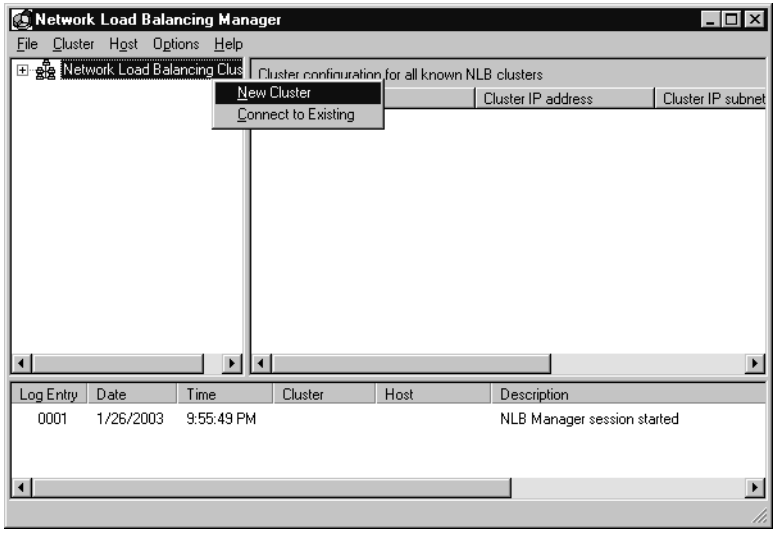

Figure 19-4 Select New Cluster to create a new NLB cluster

3. In the Cluster Parameters screen, shown in Figure 19-5, you need to enter an IP address, subnet mask, and the fully qualified domain name (FQDN) that the cluster will be known by. This IP address is a fixed IP address, so it can't be a DHCP address.

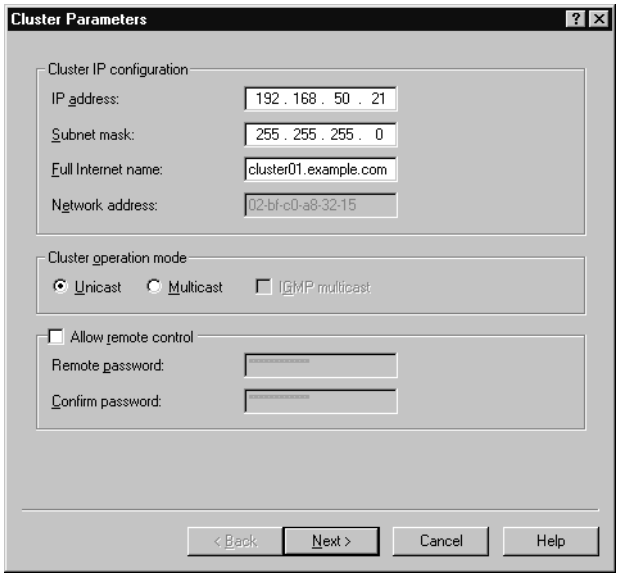

Figure 19-5 The Cluster Parameters screen

4. Select whether the cluster will be unicast or multicast and whether you will allow remote control. Then click Next.

> **Important** Allowing remote control of a cluster is a significant security issue. Before you decide to enable this, carefully consider the consequences and understand the risks. If you do decide to enable remote control of your cluster, you should enforce sound password rules on the remote password. For more information about NLB security, see *http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/clustering/nlbsecbp.mspx*.

5. If the cluster will have additional IP addresses, enter them in the Cluster IP Address screen, and then click Next.

6. You can enter port rules in the next screen, or wait to configure these after you get the cluster up and running. Port rules can be used to control the behavior of various types of TCP/IP traffic. Windows Server 2003 allows you to configure different port rules for different IP addresses. Click Next when you have configured any rules you want to configure at this point.

7. Enter the name or IP address of the first host that will be joined to the cluster in the Connect screen, shown in Figure 19-6. Click Connect to connect to the server and bring up a list of network interfaces available. Highlight the interface that will host the public traffic of the cluster (as opposed to private, node-to-node traffic).

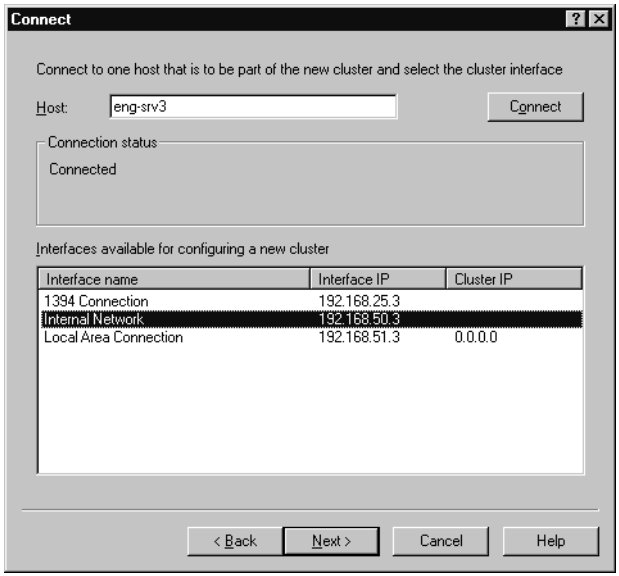

Figure 19-6 The Connect screen of the New NLB Cluster Wizard

8. Click Next to bring up the Host Parameters screen, shown in Figure 19-7. Here you set the priority for this host of the cluster and the dedicated IP address that will be used to connect to this specific server (as opposed to the cluster as a whole). This IP address must be a fixed IP address, not a DHCP address. Finally, set the initial state of this host when Windows is started.

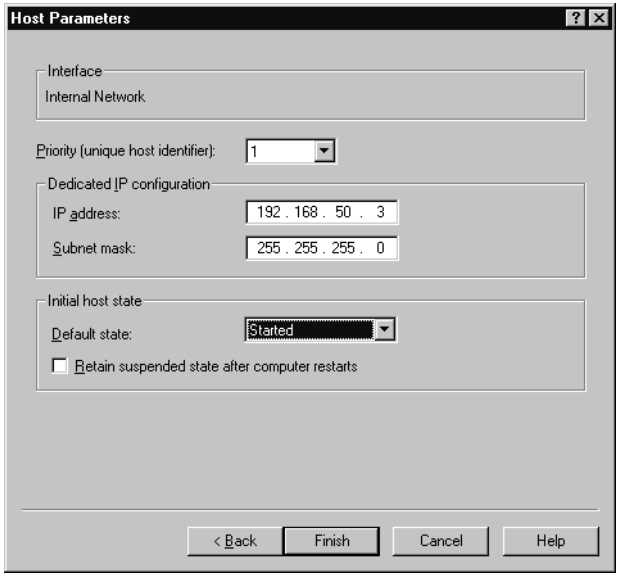

Figure 19-7 The Host Parameters screen of the New NLB Cluster Wizard

9. Click Finish to start up the NLB service and configure the server into the new cluster.

Adding a Node to an NLB Cluster

To add another node to an existing NLB cluster, follow these steps:

1. Open the Network Load Balancing Manager from the Administrative Tools folder.
2. Right-click the cluster you want to add a node to in the left pane, and select Add Host To Cluster, as shown in Figure 19-8.

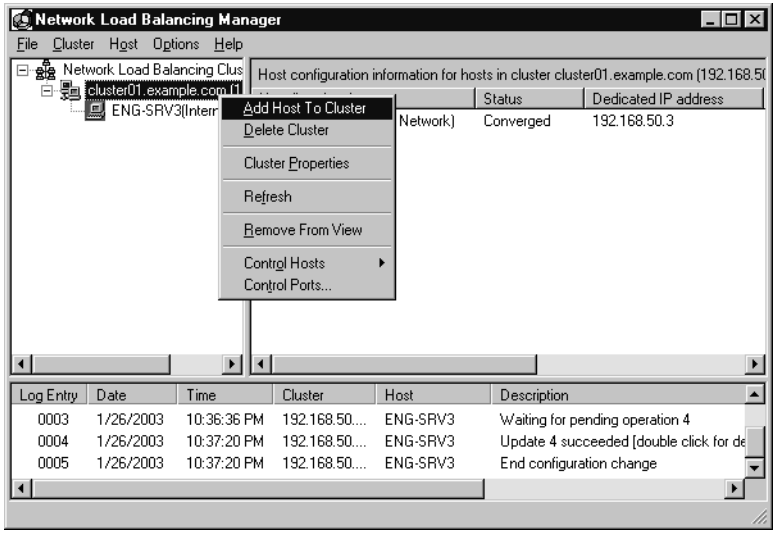

Figure 19-8 Adding a host to an existing cluster

3. Enter the name or IP address of the host that will be joined to the cluster in the Connect screen, shown previously in Figure 19-6. Click Connect to connect to the server and bring up a list of network interfaces available. Select the interface that will host the public traffic of the cluster (as opposed to private, node-to-node traffic).
4. Click Next to bring up the Host Parameters screen, shown earlier in Figure 19-7. Here you set the priority for this host of the cluster and the dedicated IP address that will be used to connect to this specific server (as opposed to the cluster as a whole). This IP address must be a fixed IP address, not a DHCP address. Finally, set the initial state of this host when Windows is started.
5. Click Finish to start up the NLB service on the new node and configure the server into the existing cluster. When the node is up and part of the cluster, it shows a status of Converged in the NLB Manager, as shown in Figure 19-9.

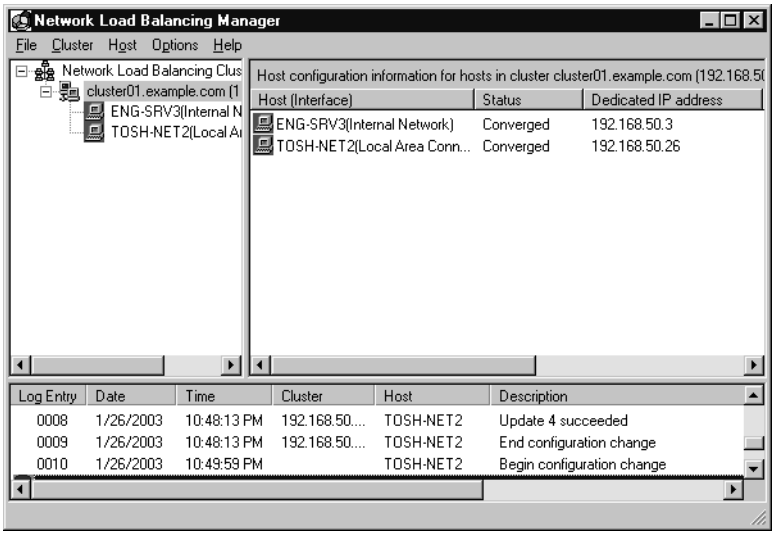

Figure 19-9 The NLB Manager shows the new node added with a status of Converged

Removing a Host from an NLB Cluster

To remove a host from an NLB cluster, follow these steps:

1. Open the Network Load Balancing Manager from the Administrative Tools folder.
2. Connect to the cluster you want to remove a node from by right-clicking Network Load Balancing Clusters in the left pane and selecting Connect To Existing.
3. Right-click the node you want to remove in the left pane, and select Delete Host.

Planning the Capacity of an NLB Cluster

In general, an NLB cluster should contain as many hosts as needed to handle the client load for the applications being run in the cluster. The exception to this would be cases in which the sole function of the cluster is to provide failover tolerance for a critical TCP/IP application—that is, when a single server can handle the load and the second server is there simply for fault tolerance.

The maximum number of hosts in a given cluster is 32. If your application requires more than 32 hosts, you can set up multiple clusters, using round-robin DNS to distribute the load among the clusters. The effective limitation, however, is likely to be the network saturation point. If you do run multiple clusters in a subnet, you should host each on its own network switch to minimize the network bottleneck.

Although fewer and more powerful servers might look cost-effective for a given application, you should consider how the failure of a server will affect the application and the remaining servers. If the remaining servers can't handle the resulting load, you could potentially have

a cascading failure, bringing down the entire application. Always provide sufficient server capacity within the cluster to handle the expected load when a single server is down. Also consider ways to limit the load to the application when there has been a failure.

When determining the expected cluster capacity, you also need to consider the application being clustered and the type of load it imposes on the cluster. Plan your servers according to where the limitation and stress will be greatest. Web serving and FTP applications are input/output (I/O) intensive, whereas Terminal Services can be very CPU intensive, depending on the types of applications your user community uses.

Providing Fault Tolerance

Although NLB clusters provide overall fault tolerance for your TCP/IP application, they are not a complete solution for all possible failures. Because they are "shared nothing" clusters, there is always some data lag between servers. For fully fault-tolerant, high-availability clustering that can run any application, you should probably use server clustering, which provides the greatest level of fault tolerance.

One thing you can do to improve the overall fault tolerance of the cluster is to make the hard disks fault tolerant, whether physically attached to the server or as Network-Attached Storage (NAS). Both hardware and software RAID solutions are viable options for improving the fault tolerance of an NLB cluster. For more on RAID and fault tolerance in general, see Chapter 18 and Chapter 38.

Optimizing an NLB Cluster

Optimizing an NLB cluster calls for clearly understanding where the bottleneck in your clustered application is likely to be. An application such as a Web front end that is essentially a file server, for example, tends to be a heavy user of both disk I/O and network bandwidth, and such an application can be a RAM hog if you're going to do effective caching. Terminal Services, on the other hand, can put a heavy load on the CPU, and to a somewhat lesser extent, on RAM, depending on your user community. Focus your optimization efforts on the bottleneck and you'll get the most gain for your effort.

One area that can be a problem is running an NLB cluster in a switched environment without planning your network load carefully. If each of the servers in your cluster is connected to a different switched port, you can easily end up flooding your switched ports because every client request to the cluster passes through all switched ports to which a member of the cluster is attached. Running in multicast mode can exacerbate the problem. If you're running in a switched environment, you should follow these guidelines:

1. Use a top-quality hub to connect the servers in the cluster to one another, and uplink the hub to a single switched port. If you do use switches, separate each cluster onto its own VLAN.

2. Use unicast mode. If you enabled multicast mode during setup, change it. (You'll need to change this on all servers in the cluster.) It is possible to use multicast mode, but this requires enabling Internet Group Multicast Protocol (IGMP) support, introduced in Windows Server 2003. Given the other limitations multicast mode, however, unicast is preferred.
3. Edit the registry on each of the hosts in the cluster, changing the following key from the default parameter of 1 to 0:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\WLBS
  \Parameters\MaskSourceMAC
```

   This change allows the switch to tell which MAC address is really the source of traffic, helping it to do its switching job properly. You'll need to restart the servers after making this change.

Server Clusters

A server cluster is a group of independent nodes that work together as a single system. They share a common cluster database that enables recovery in the event of the failure of any node. A traditional server cluster uses a jointly connected resource, generally a disk array on a shared SCSI bus or Fibre Channel, which is available to all nodes in the cluster. Each Windows Server 2003 Enterprise Edition node in the cluster must have access to the array, and each node in the cluster must be able to communicate at all times with the other nodes in the cluster.

Windows Server 2003 supports server clusters only on machines running Enterprise Edition or Datacenter Edition. Both editions support up to eight node clusters and can be configured in three different models: single node clusters, single quorum device clusters, and majority node set clusters. We focus on single quorum device clusters in this chapter. Single node clusters are primarily used for creating virtual servers and for proof of concept and development of cluster-aware applications. Majority node set server clusters require specialized support from both the hardware and software vendors involved.

Server Cluster Concepts

To understand and implement server clusters, it is important to understand several new concepts and their ramifications, as well as specialized meanings for certain terms.

Networks (Interconnects)

A cluster has two distinct types of networks: the *private network* that's used to maintain communications between nodes in the cluster and the *public network* that clients of the cluster use to connect to the services of the cluster. Each of these networks can share the

same network card and physical network cabling, but it is a good practice to keep them separate. This gives you an alternate path for interconnection between the nodes of the cluster. Because the interconnect between the nodes of a cluster is a potential single point of failure, it should always be redundant. The cluster service uses all available networks, both private and public, to maintain communications between nodes.

Real World Always Have at Least Two Interconnects

If you have only a single method of communication in a cluster, the failure of that interconnect has a 50 percent chance (in a two-node cluster) of causing the entire cluster to become unavailable to its clients—hardly why you opted for a highly available technology like clustering. Here's what happens when the nodes of a cluster can no longer communicate. When the communications fail, each node recognizes that it is no longer able to talk to the other nodes of the cluster and decides that the other nodes in the cluster have failed. Each node therefore attempts to take over the functions of the cluster by itself. The nodes are "partitioned," and as each node attempts to enable itself to take over the functions of the entire cluster, it starts by trying to gain control of the quorum resource (discussed later in the "Types of Resources" section) and, therefore, the shared disk on which the quorum resides. Because only one node is able to gain control of the quorum resource, the other nodes are automatically shut down while the single node attempts to maintain the processes of the cluster. However, because any given node has an equal chance of gaining control of the quorum resource, there's a 50 percent chance in a two-node cluster that the node with a failed network card wins, leaving all the services of the cluster unavailable.

Nodes

A *node* is a member of a server cluster. It must be running Windows Server 2003, Enterprise Edition or Windows Server 2003, Datacenter Edition, and Windows Clustering. It must also be running TCP/IP, be connected to the shared cluster storage device, and have at least one network interconnect to the other nodes in the cluster.

Groups

Groups are the units of failover. Each group contains one or more resources. Should any of the resources within the group fail, all fail over together according to the failover policy defined for the group. A group can be owned by only one node at a time. All resources within the group run on the same node. If a resource within the group fails and must be moved to an alternate node, all other resources in that group must be moved as well. When the cause of failure on the originating node is resolved, the group falls back to its original location, based on the failback policy for the group.

Resources

Any physical or logical entity that can be brought online or offline can be a server cluster resource. It must be able to be owned by only one node at a time and will be managed as part of the cluster. The *quorum resource* is a special resource that serves as the repository of the configuration data of the cluster and the recovery logs that allow recovery of the cluster in the event of a failure. The quorum resource must be able to be controlled by a single node, it must provide physical storage for the recovery logs and cluster database, and it must use the NTFS file system. The only resource type supported for a quorum resource in single quorum device clustering is the Physical Disk resource as shipped with Windows Server 2003 (which along with other resource types are described in the next section), but it is possible that other quorum resource types will be developed and certified by third parties.

Types of Resources

Windows Server 2003 Enterprise Edition includes several resource types; the sections that follow examine each of these resource types and the role they play in a server cluster. The available cluster resource types are as follows:

- Physical Disk
- Dynamic Host Configuration Protocol (DHCP)
- Windows Internet Naming Service (WINS)
- Print Spooler
- File Share
- Internet Protocol Address
- Local Quorum
- Majority Node Set
- Network Name
- Generic Application
- Generic Script
- Generic Service
- Volume Shadow Copy Service Task

Physical Disk

The Physical Disk resource type is the central resource type required as a minimum for all server clusters. It is used for the quorum resource that controls which node in the cluster

is in control of all other resources. The Physical Disk resource type is used to manage a shared cluster storage device. It has the same drive letter on all cluster servers.

DHCP and WINS

The DHCP service provides IP addresses and various other TCP/IP settings to clients, and WINS provides dynamic resolution of NetBIOS names to IP addresses. Both can be run as a resource of the cluster, providing for high availability of these critical services to network clients. For failover to work correctly, the DHCP and WINS databases must reside on the shared cluster storage.

Print Spooler

The Print Spooler resource type lets you cluster print services, making them fault tolerant and saving a tremendous number of help desk calls when the print server fails. It also ameliorates the problem of people simply clicking Print over and over when there's a problem, resulting in a long and repetitious print queue.

To be clustered, a printer must be connected to the server through the network. Obviously, you can't connect the printer to a local port such as a parallel or Universal Serial Bus (USB) port directly attached to one of the nodes of the cluster. The client can address the printer either by name or by IP address, just as it would a nonclustered printer on the network.

In the event of a failover, all jobs that are currently spooled to the printer are restarted. Jobs that are in the process of spooling from the client are discarded.

File Share

You can use a server cluster to provide a high-availability file server using the File Share resource type. The File Share resource type lets you manage your shared file systems in three different ways:

- As a standard file share with only the top-level folder visible as a share name.
- As shared subfolders, where the top-level folder and each of its immediate subfolders are shared with separate names. This approach makes it extremely easy to manage users' home directories, for example.
- As a standalone Distributed file system (Dfs) root. You cannot, however, use a cluster server File Share resource as part of a fault-tolerant Dfs root.

Internet Protocol Address and Network Name

The Internet Protocol Address resource type is used to manage the IP addresses of the cluster. When an Internet Protocol Address resource is combined with a Network Name resource and one or more applications, you can create a *virtual server*. Virtual servers allow clients to continue to use the same name to access the cluster even after a failover

has occurred. No client-side management is required because, from the client perspective, the virtual server is unchanged.

Local Quorum

The Local Quorum resource type is used to manage the system disk on the local node of a single node server cluster. The Local Quorum resource type cannot fail over to another node.

Majority Node Set

The Majority Node Set resource type is used to manage cluster configuration data that might or might not reside on a cluster storage device. It is used to ensure that the data remains consistent across nodes that may be geographically dispersed. Only a single Majority Node Set resource can exist in a server cluster.

Generic Application

The Generic Application resource type allows you to manage regular, cluster-unaware applications in the cluster. A cluster-unaware application that is to be used in a cluster must, at a minimum:

- Be able to store its data in a configurable location
- Use TCP/IP to connect to clients
- Have clients that can reconnect in the event of an intermittent network failure

When you install a generic, cluster-unaware application, you have two choices: you can install it onto the shared cluster storage, or you can install it individually on each node of the cluster. The first method is certainly easier because you install the application only once for the whole cluster. However, if you use this method you won't be able to perform a rolling upgrade of the application, because it appears only once. (A rolling upgrade is an upgrade of the application in which the workload is moved to one server while the application on the other server is upgraded and then the roles are reversed to upgrade the first server.)

To give yourself the ability to perform rolling upgrades on the application, you need to install a copy onto each node of the cluster. You need to place it in the same folder and path on each node. This method uses more disk space than installing onto the shared cluster storage, but it permits you to perform rolling upgrades, upgrading each node of the cluster separately.

Generic Script

Similar to the Generic Application resource, the Generic Script resource type is used to manage operating system scripts as a cluster resource. The Generic Script resource type provides limited functionality.

Generic Service

Finally, server clusters support one additional type of resource—the Generic Service resource. This is the most basic resource type, but it does allow you to manage your Windows Server 2003 services as a cluster resource.

Volume Shadow Copy Service Task

The Volume Shadow Copy Service Task resource type allows you to create jobs in the Scheduled Task folder that will be run against whatever node is currently hosting a particular resource group, allowing the task to fail over with the resource. As shipped, this resource type is used only to support Shadow Copies of Shared Folders in a server cluster.

Defining Failover and Failback

Windows Server 2003 server clusters allow you to define the failover and failback (sometimes referred to as fallback) policies for each group or virtual server. This ability enables you to tune the exact behavior of each application or group of applications to balance the need for high availability against the overall resources available to the cluster in a failure situation. Also, when the failed node becomes available again, your failback policy determines whether the failed resource is immediately returned to the restored node, maintained at the failed-over node, or migrated back to the restored node at some predetermined point in the future. These options allow you to plan for the disruption caused when a shift in node ownership occurs, limiting the impact by timing it for off-hours.

Configuring a Server Cluster

When planning your server cluster, you'll need to think ahead to what your goal is for the cluster and what you can reasonably expect from it. Server clusters provide for extremely high availability and resource load balancing, but you need to make sure your hardware, applications, and policies are appropriate.

High Availability with Load Balancing

The most common cluster configuration is static load balancing. In this scenario, the cluster is configured so that some applications or resources are normally hosted on one node whereas others are normally hosted on another node. If one node fails, the applications or resources on the failed node fail over to another node, providing high availability of your resources in the event of failure and balancing the load across the cluster during normal operation. The limitation of this configuration is that in the event of a failure, your applications will all attempt to run on fewer nodes, and you need to implement procedures either to limit the load by reducing performance or availability, or to not provide some less critical services during a failure. Another possibility for managing the reduced

load-carrying capacity during a failure scenario is to have "at risk" users and applications that can be shut off or "shed" during periods of reduced capacity, much like power companies do during peak load periods when capacity is exceeded.

It's important to quickly take steps to manage load during periods of failure when you configure your cluster for static load balancing. Failure to shed load can lead to catastrophic failure, or such extreme slowdown as to simulate it, and then no one will have access to the cluster's resources and applications.

Maximum Availability Without Load Balancing

The cluster configuration with the highest availability and reliability for critical applications is to run one node of the cluster as a hot spare. This scenario requires that the hot spare node be sufficiently powerful to run the entire load of any other node in the cluster. You then configure all the applications and resources to run on the other nodes, with the one node sitting idle. In the event of failure on one of the primary nodes, the applications fail over to the idle node and continue with full capability. After the primary node is back online it can continue as the new hot spare, or you can force the applications back to the primary node, depending on the needs of your environment.

This scenario provides full and complete fault tolerance in the event of the failure of one of the nodes, but it has the greatest hardware cost. It also does not provide for full and complete fault tolerance in the event of multiple node failures—that would take essentially one hot spare for each primary node. Use this clustering configuration only where your applications or resources are critical and you can afford the extra hardware expense far more than any limits to the load in case of a failure.

Partial Failover (Load Shedding)

Another cluster configuration is called load shedding or partial failover. In this configuration, critical applications and resources are designed to fail over to the other nodes in the cluster in the event of a failure, but noncritical applications and resources are unavailable until the cluster is back to full functionality. The critical resources and applications are thus protected in a failure situation, but noncritical ones simply run as though they were on a stand-alone server.

In this configuration, you might, depending on capacity and load conditions, have to configure the noncritical applications and resources on all nodes to be unavailable in the event of a failure on other nodes. This allows you to maintain a high level of performance and availability for your most critical applications while shedding the load from less critical applications and services when necessary. This strategy can be very effective when you must, for example, service certain critical applications or users under any and all circumstances but can allow other applications and users with a lower priority to temporarily fail.

Virtual Server Only

You can create a server cluster that has only a single node, which allows you to take advantage of the virtual server concept to simplify the management and look of the resources on your network. For example, the File Share resource lets you create automatic subdirectory shares of your primary share and control their visibility, a perfect way to handle users' home directories. Having a single node doesn't give you any additional protection against failure or any additional load balancing over that provided by simply running a single standalone server, but it allows you to easily manage groups of resources as a virtual server.

This scenario is an effective way to stage an implementation. You create the initial virtual server, putting your most important resources on it in a limited fashion. Then, when you're ready, you add another node to the server cluster and define your failover and failback policies, giving you a high-availability environment with minimal disruption to your user community. In this scenario, you can space hardware purchases over a longer period while providing services in a controlled test environment.

Planning the Capacity of a Server Cluster

Capacity planning for a server cluster can be a complicated process. You need to thoroughly understand the applications that will be running on your cluster and make some hard decisions about exactly which applications you can live without and which ones must be maintained under all circumstances. You'll also need a clear understanding of the interdependencies of the resources and applications you'll be supporting.

The first step is to quantify your groups or virtual servers. Applications and resources that are in the same group will fail over together onto the same server. This means you'll need to plan out which applications are dependent on each other and will need to function together. Make a comprehensive list of all applications in your environment, and then determine which ones need to fail over and which ones can be allowed to simply fail but still should be run on a virtual server.

Next, determine the dependencies of the applications and the resources they need to function. This allows you to group dependent applications and resources in the same group or virtual server. Keep in mind that a resource can't span groups, so if multiple applications depend on a resource, such as a Web server, they must all reside in the same group or on the same virtual server as the Web server and thus share the same failover and failback policies.

A useful mechanism for getting a handle on your dependencies is to list all your applications and resources and draw a dependency tree for each major application or resource. This helps you visualize not only the resources that your application is

directly dependent on, but also the second-hand and third-hand dependencies that might not be obvious at first glance. For example, a cluster that is used as a high-availability file server uses the File Share resource. And it makes perfect sense that this File Share resource is dependent on the Physical Disk resource. It's also dependent on the Network Name resource. However, the Network Name resource is dependent on the IP Address resource. Thus, although the File Share resource isn't directly dependent on the IP Address resource, when you draw the dependency tree you will see that they all need to reside in the same group or on the same virtual server. Figure 19-10 illustrates this dependency tree.

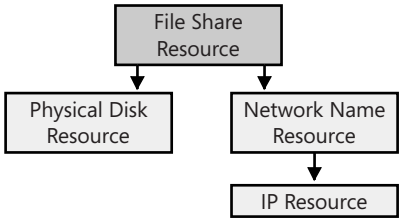

Figure 19-10 The dependency tree for a File Share resource

Finally, as you're determining your cluster capacity, you need to plan for the effect of a failover. Each server must have sufficient capacity to handle the additional load imposed on it when a node fails and it is required to run the applications or resources owned by the failed node.

The disk capacity for the shared cluster storage must be sufficient to handle all the applications that will be running in the cluster and to provide the storage that the cluster itself requires for the quorum resource. Be sure to provide enough RAM and CPU capacity on each node of the cluster so that the failure of one node won't overload the other node to the point that it too fails. This possibility can also be managed to some extent by determining your real service requirements for different applications and user communities and reducing the performance or capacity of those that are less essential during a failure. However, such planned load shedding might not be sufficient and frequently takes a significant amount of time to accomplish, so give yourself some margin to handle that initial surge during failover.

Creating a Server Cluster

Once you've thoroughly researched and planned your implementation of server clusters, you're ready to actually create the cluster. The mechanism to create and manage server clusters is the Cluster Administrator application, part of the Administrative Tools folder.

New Server Cluster

To create a new server cluster, follow these steps:

1. Open the Cluster Administrator from the Administrative Tools folder. Select Create New Cluster from the drop-down list in the Open Connection To Cluster dialog box, as shown in Figure 19-11.

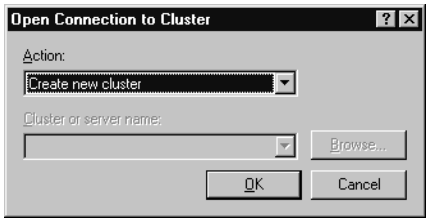

Figure 19-11 The Open Connection To Cluster dialog box

2. Click OK to launch the New Server Cluster Wizard, shown in Figure 19-12. The New Server Cluster Wizard walks you through testing to see if the cluster can be successfully created. It also gives you an opportunity to correct issues it discovers during the test, and then actually creates the cluster.

Figure 19-12 The New Server Cluster Wizard

3. Click Next to bring up the Cluster Name And Domain page, as shown in Figure 19-13. The domain is generally already filled in with the current domain. Fill in the name for the cluster. You can make this a name that means something to you, as opposed to your user community, because you'll likely be creating virtual servers for it.

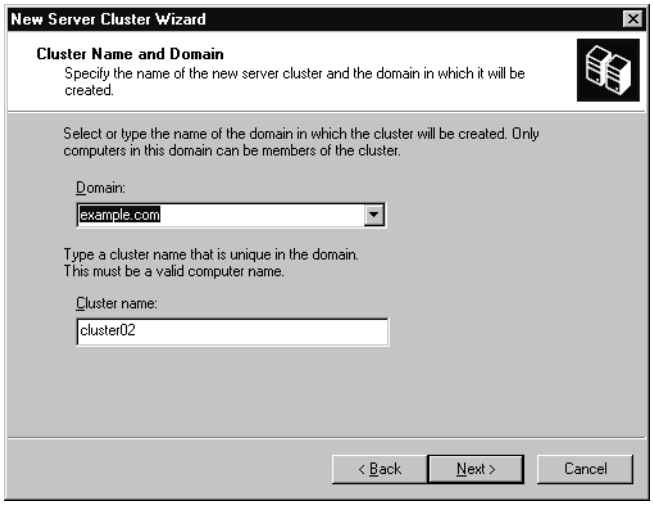

Figure 19-13 The Cluster Name And Domain page of the New Server Cluster Wizard

4. Click Next to bring up the Select Computer page, shown in Figure 19-14. Enter the name of the computer that will be the first computer in the new cluster in the Computer Name field.

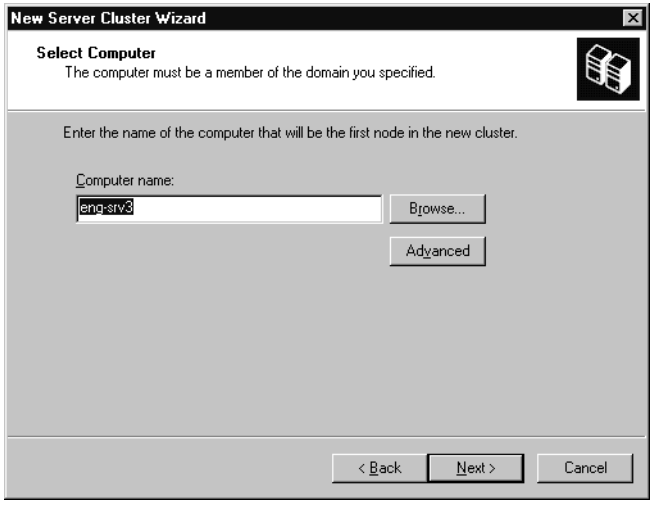

Figure 19-14 The Select Computer page of the New Server Cluster Wizard

5. Click Next to bring up the Analyzing Configuration page. The wizard automatically analyzes the configuration and highlights any problems, as shown in Figure 19-15. If the bar is green, the problems it found are nonfatal and you could go ahead and create the cluster. However, you should attempt to correct any problems before proceeding.

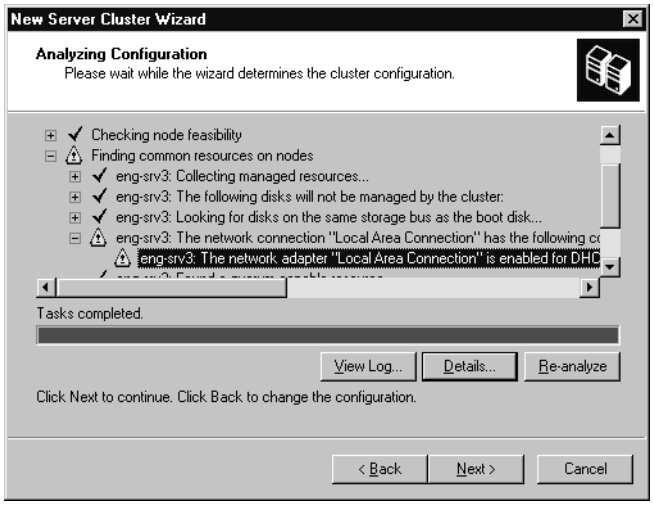

Figure 19-15 The Analyzing Configuration page of the New Server Cluster Wizard

6. To view details on the problems found, click View Log. A typical problem is shown in Figure 19-16. You can correct the problem (in this case, one of the network adapters was configured for DHCP, a nonrecommended configuration) and then click Re-analyze to run the analysis again.

7. Once the Analyzing Configuration Wizard gives you a clean bill of health, click Next to open the IP Address page, shown in Figure 19-17. Enter the IP address that will be used by clustering management tools to connect to the cluster.

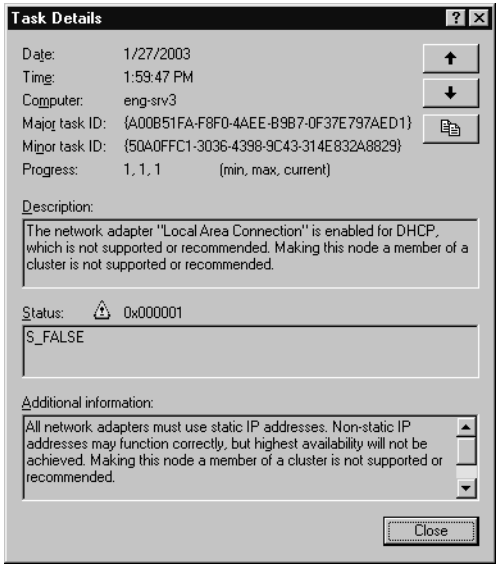

Figure 19-16 The Task Details page, showing that one adapter is configured for DHCP

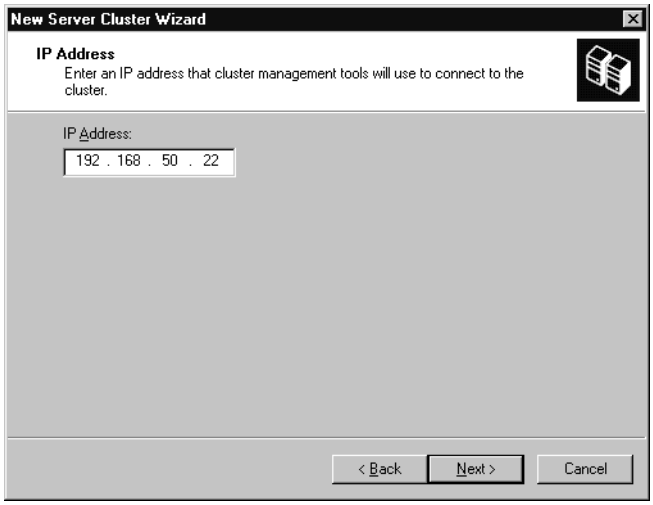

Figure 19-17 The IP Address page of the New Server Cluster Wizard

8. Click Next to bring up the Cluster Service Account page shown in Figure 19-18. This can be an existing account or a new account. The account will be given local administrative privileges on all nodes of the cluster. Click Next.

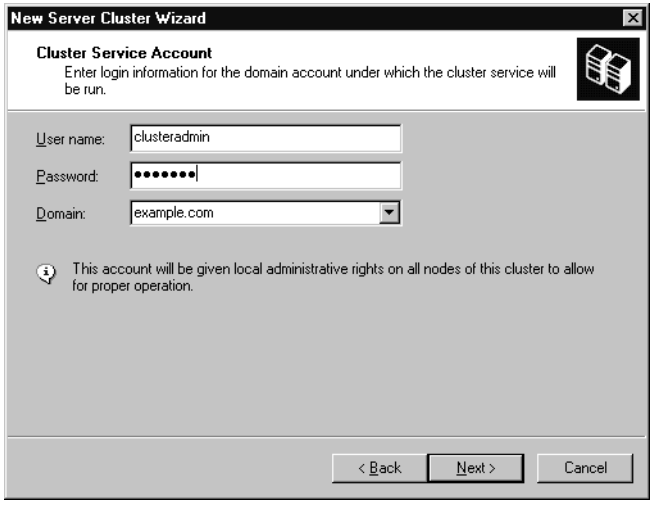

Figure 19-18 The Cluster Service Account page of the New Server Cluster Wizard

9. The final confirmation page is shown in Figure 19-19. Spend a moment here to verify that this is really what you want to do and that everything agrees with your checklist. You can go back and fix anything before continuing, if necessary, so take the time now.

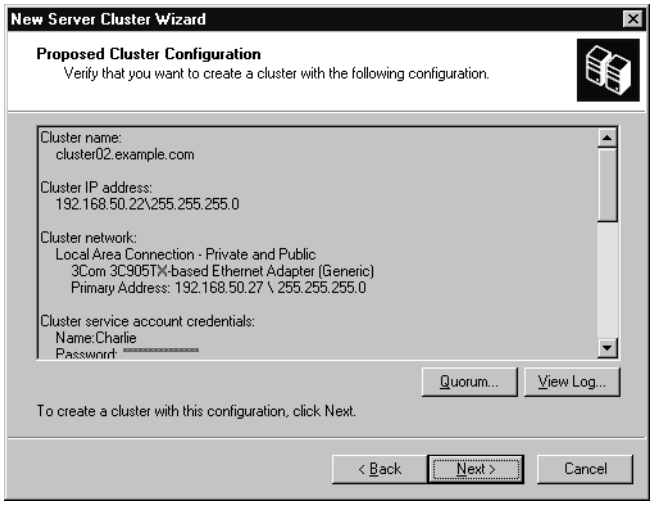

Figure 19-19 The Proposed Cluster Configuration page of the New Server Cluster Wizard

10. When you're ready, click Next to start creating the cluster. When the process is complete, you'll see a status page as shown in Figure 19-20. Click View Log to see a log of the process, or click Details to see more detailed steps than those shown. If there were problems, you'll be able to go back and correct them and try again. Click Next.

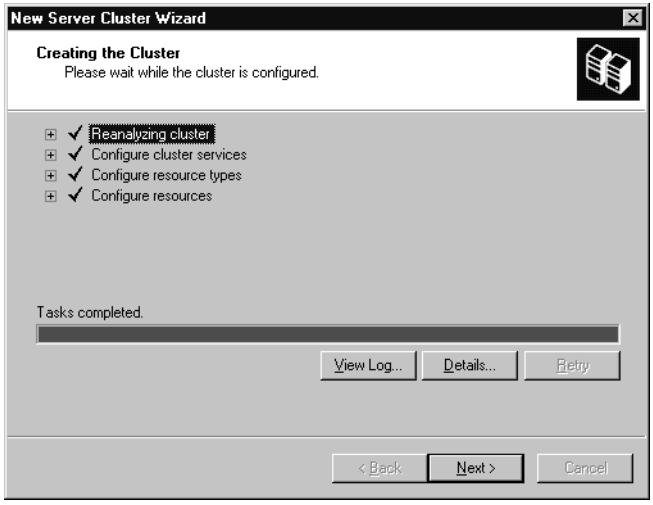

Figure 19-20 The Creating The Cluster page of the New Server Cluster Wizard

11. This brings you to the final page of the New Server Cluster Wizard. You can view the log from here by clicking View Log or change from Local Quorum to Majority Node Set by clicking the Quorum button. Click Finish and the New Server Cluster

Wizard exits, leaving you in the Cluster Administrator application, as shown in Figure 19-21.

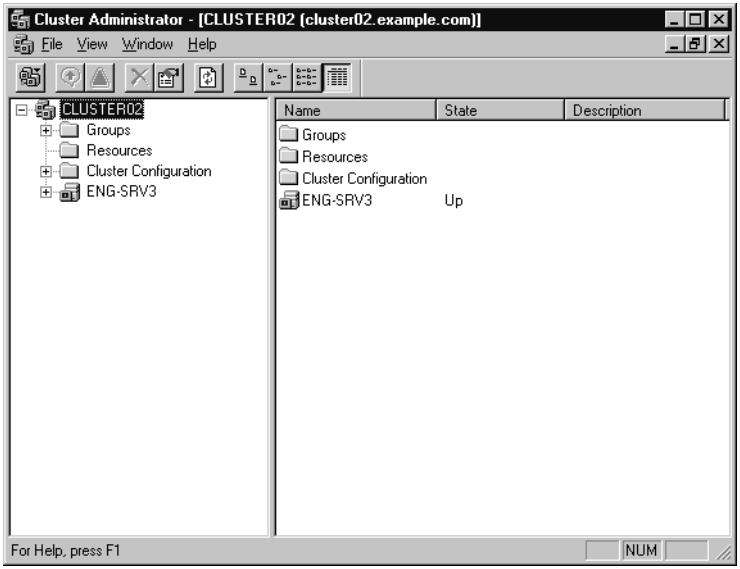

Figure 19-21 The Cluster Administrator application

Creating a Clustered Resource

Once you have your cluster created, you can take advantage of the management capabilities of Cluster Administrator to create cluster resources. We'll walk through the steps to create a File Share cluster resource in a new group on a virtual server called HOME. Referring to Figure 19-10, you'll see the list of dependencies we need to deal with. Although we could put these resources in the main Cluster group, we prefer to group items into more logical units, especially because failover policies are controlled at the group level. Therefore, to create our File Share resource, we'll need to do the following:

- Create a group to hold the necessary resources
- Create a Physical Disk resource
- Create an IP Address resource
- Create a Network Name resource
- Create the File Share resource

New Cluster Group

To create a new cluster group, follow these steps:

1. Open the Cluster Administrator from the Administrative Tools folder, and connect to the cluster where you will be creating the resource.

2. Right-click the Active Groups folder of the server that will host the File Share resource, and select Group from the New menu, as shown in Figure 19-22.

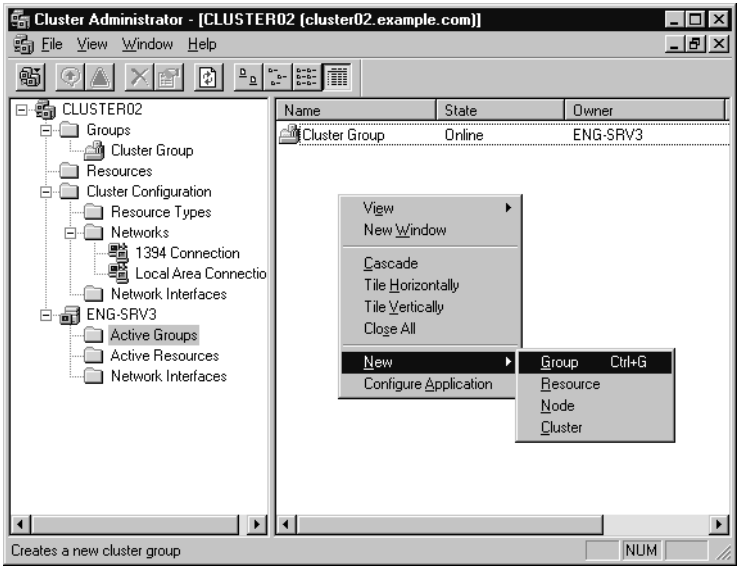

Figure 19-22 The shortcut menu to create a new group

3. This opens the New Group Wizard shown in Figure 19-23. Give your new cluster group an appropriate name and description.

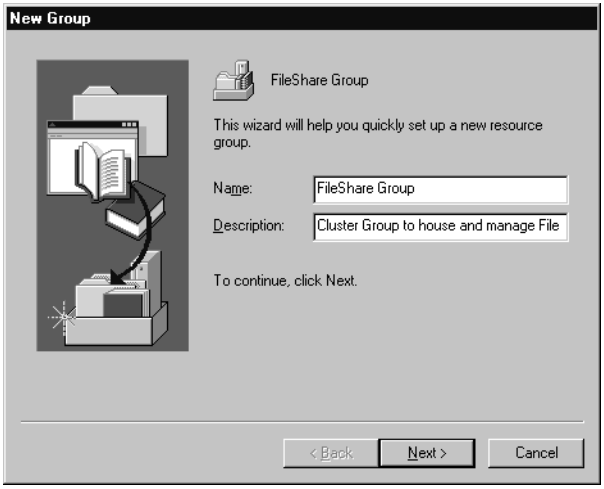

Figure 19-23 The New Group Wizard

4. Click Next to bring up the Preferred Owners dialog box shown in Figure 19-24. This allows you to control which nodes are the preferred owners of this share, and

the order of preference. Select the nodes to be used for this group, and click Add to move them to the right pane. Use the Move Up and Move Down buttons to arrange their order of precedence.

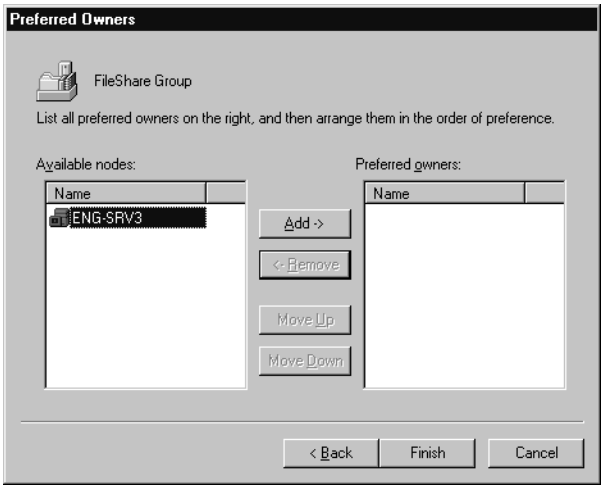

Figure 19-24 The Preferred Owners dialog box of the New Group Wizard

5. Click Finish to create the group. The group is created and is initially offline, because it has no active resources associated with it.

New Physical Disk Resource

To create a new Physical Disk resource, continue with the following steps:

1. Right-click the group just created and select Resource from the New menu to open the New Resource Wizard, shown in Figure 19-25.

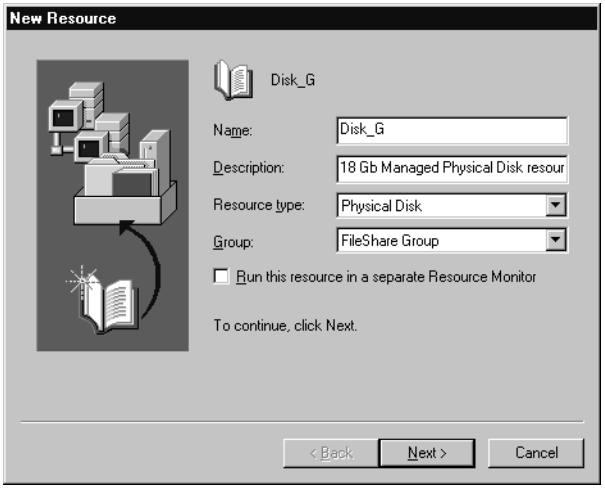

Figure 19-25 The New Resource Wizard

2. Fill in the Name and Description fields, and select Physical Disk from the Resource Type drop-down list. The Group should be the one you just created.
3. Click Next to open the Possible Owners page, as shown in Figure 19-26. Specify which machines in the cluster can host this resource.

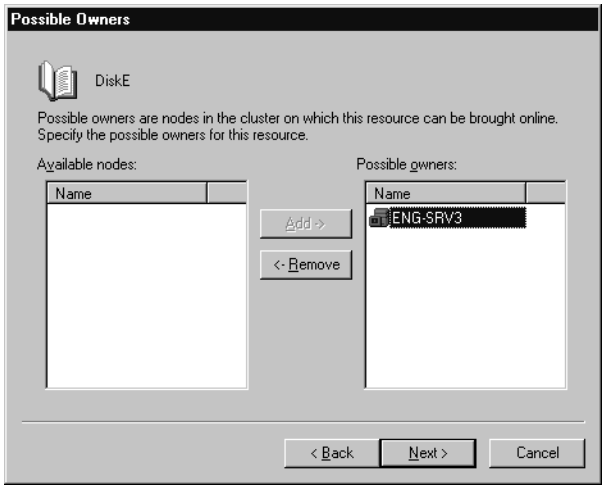

Figure 19-26 The Possible Owners page of the New Resource Wizard

4. Click Next to open the Dependencies page. This will be blank because this is the first resource in this group.
5. Click Next to open the Disk Parameters page, shown in Figure 19-27. The Disk drop-down list will include all Physical Disk resources that can be managed by the cluster service.

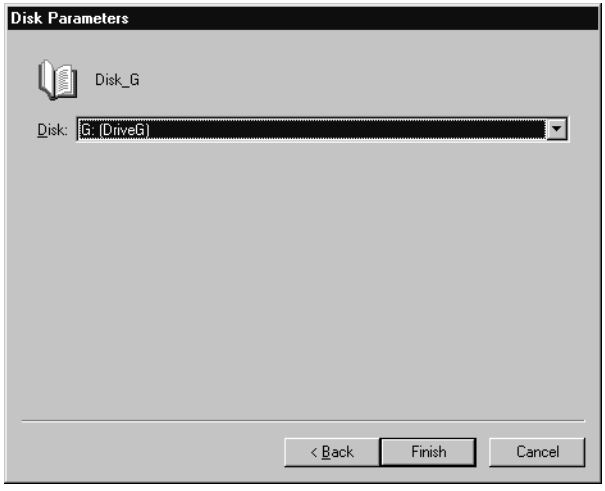

Figure 19-27 The Disk Parameters page of the New Resource Wizard

> **Note** The cluster service can manage only basic disks, not dynamic disks, and all partitions on the disk must be formatted with NTFS. Windows Server 2003 server clusters do support volume mount points.

6. Select the disk that will be the Physical Disk resource, and click Finish to create the resource.

New IP Address Resource

To add a new IP address resource, continue with the following steps:

1. Right-click the group just created and select Resource from the New menu to open the New Resource Wizard, shown earlier in Figure 19-25.
2. Fill in the Name and Description fields, and select File Share from the Resource Type drop-down list. The Group should be the one you just created.
3. Click Next to open the Possible Owners page, shown earlier in Figure 19-26. Specify which machines in the cluster can host this resource.
4. Click Next to open the Dependencies page. This will have the Physical Disk resource we just created, but referring to our dependency tree in Figure 19-10, we see that there is no dependency for the IP Address resource type.
5. Click Next to open the TCP/IP Address Parameters page, shown in Figure 19-28. Fill in the IP address and parameters that will be used for this share.

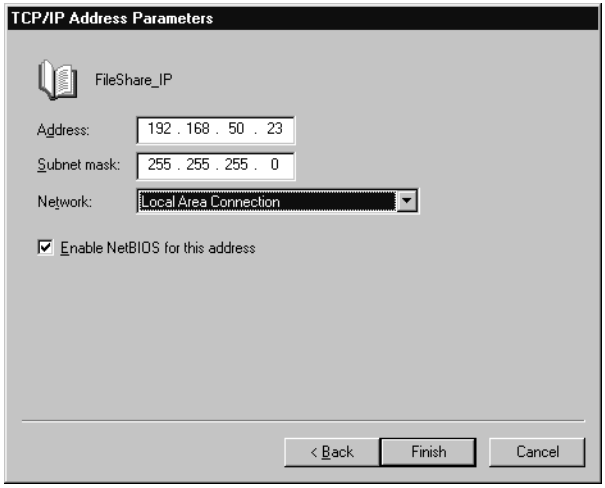

Figure 19-28 The TCP/IP Address Parameters page of the New Resource Wizard

6. Click Finish to create the IP Address resource.

New Network Name Resource

To add a new network name resource, continue with the following steps:

1. Right-click the group just created and select Resource from the New menu to open the New Resource Wizard shown earlier in Figure 19-25.
2. Fill in the Name and Description fields, and select Network Name from the Resource Type drop-down list. The Group should be the one you just created.
3. Click Next to open the Possible Owners page, shown earlier in Figure 19-26. Specify which machines in the cluster can host this resource.
4. Click Next to open the Dependencies page, shown in Figure 19-29. We'll now see both the Physical Disk and IP Address resources on the list of available resources. By looking at the dependency tree, we see that the Network Name resource has a dependency on the IP Address resource, so select the IP Address resource in the left plane and click Add to move it to the right dependencies pane.

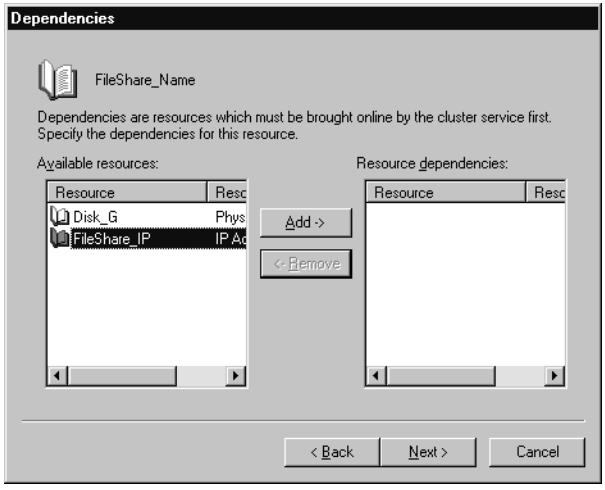

Figure 19-29 The Dependencies page

5. Click Next to open the Network Name Parameters page, and enter the name for the virtual server.
6. Click Finish to create the Network Name resource.

New File Share Resource

Finally, we're ready to create the File Share resource, because we've made all the dependencies.

1. Right-click the group just created and select Resource from the New menu to open the New Resource Wizard shown earlier in Figure 19-25.

2. Fill in the Name and Description fields, and select IP Address from the Resource Type drop-down list. The Group should be the one you just created.
3. Click Next to open up the Possible Owners page, shown earlier in Figure 19-26. Specify which machines in the cluster can host this resource.
4. Click Next to open the Dependencies page. We'll now see all three of the resources we've just created. We know that all of them are required for the File Share resource to work, so we'll move them all to the rightmost dependency pane.
5. Click Next to open the File Share Parameters page shown in Figure 19-30.

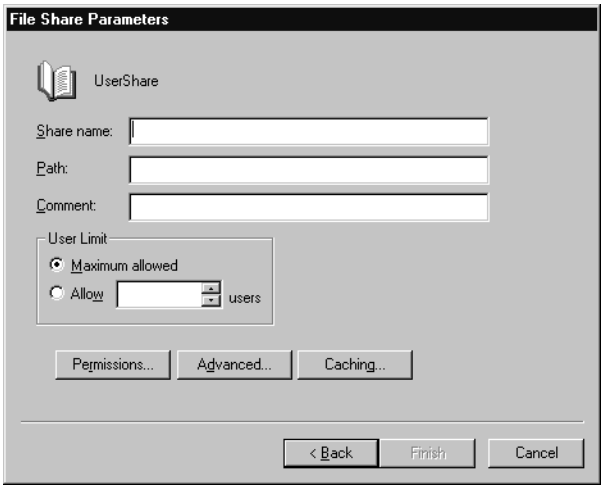

Figure 19-30 The File Share Parameters page

6. Fill in the Share Name and Path fields, and add a description in the Comment text box. If you click Finish now, you'll end up with a simple File Share.
7. Click Advanced to open the Advanced File Share Properties dialog box shown in Figure 19-31. Select Share Subdirectories and Hide Subdirectory Shares, and click OK.

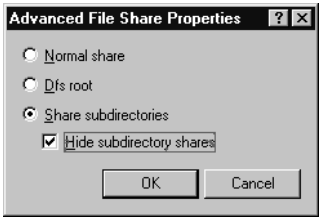

Figure 19-31 Advanced File Share Properties dialog box

8. Click Finish to create the File Share resource, and click OK to acknowledge the success.

9. Finally, right-click the group you created and select Bring Online to make the resource actually online and available.

Note The File Share resource will, by default, set the file system permissions to Read Only. Click Permissions in the File Share Parameters screen to change that as required.

Compute Clusters

As this edition of the book is being written, Microsoft is beta testing a new kind of Windows Server clustering—High Performance Computing (HPC) clusters, also called compute clusters. Unlike NLB or server clusters, compute clusters are not designed to provide high availability for critical applications, but rather to distribute highly parallel and complex computing tasks across multiple nodes. Windows Compute Cluster Server 2003 (CCS) enables super-computer functionality on the desktop.

CCS is a combination of a special version of Windows Server 2003 x64 Edition; the Compute Cluster Edition (CCE); and a package of interfaces, management tools, and utilities known as the Compute Cluster Pack. The Compute Cluster Pack is available separately and can be installed on any x64 Edition of Windows Server 2003, including R2 versions. CCE is at the SP1 level and does not support installation of R2.

Note Windows Compute Cluster Server 2003 is supported *only* on x64 Editions of Windows Server 2003. It is not available on 32-bit Windows Server 2003, nor on Itanium 64-bit editions.

CCS supports configurations that have one, two, or three NICs per node. The preferred configuration is a head node with a public (internal network LAN) interface, a private intra-cluster communications interface, and a high-speed Message Passing Interface (MPI). Each compute node in the cluster would have at least a private communications interface and the MPI interface. Figure 19-32 shows this topology.

CCS includes Remote Installation Services (RIS) that allows the easy setup and deployment of compute nodes on demand. Once the head node is created and configured, individual compute nodes are simply connected to the network and powered up. RIS then deploys Windows Server 2003 Compute Cluster Edition to the new node and configures it to be part of the compute cluster.

CCS includes the Microsoft Message Passing Interface (MS MPI), a highly compatible implementation of the Argonne National Labs MPICH2 specification. Because the MS MPI implementation is completely compatible with MPICH2 at the API level, existing HPC applications that use MPICH2 will easily migrate to CCS.

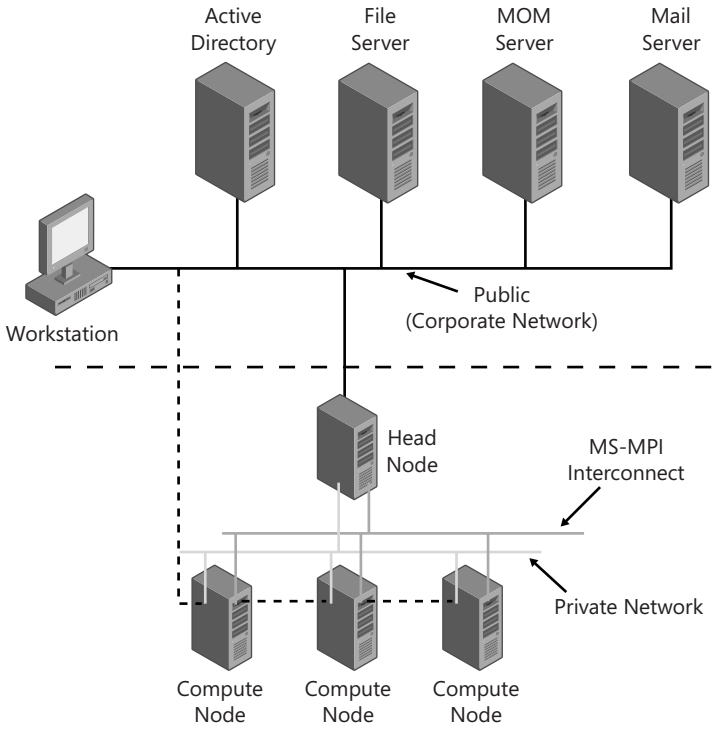

Figure 19-32 Windows Compute Cluster Server 2003 Topology

More Info For additional information on CCS, including details of MPI, migration of existing parallel applications, and parallel debugging, see *http://www.microsoft.com/windowsserver2003/ccs/default.mspx*.

Summary

Windows Server 2003, Enterprise Edition provides two high-availability clustering models: Network Load Balancing clusters (formerly known as the Windows Load Balancing Service) and server clusters. Clusters provide a highly available and scalable environment. Network Load Balancing clusters use standard hardware to distribute TCP/IP applications across a cluster. Server clusters use specialized shared disk resources to provide failover and static load balancing for a variety of applications. A new type of Windows Server 2003 cluster, the compute cluster, supports high-performance computing and highly parallel computing tasks. The next chapter covers configuring your storage as well as planning for fault tolerance and flexibility in managing your storage needs.