MICROSOFT® PROFESSIONAL EDITIONS

For Windows NT Server Version 4.0

Microsoft® Windows NT® Server Resource Guide

Technical Information and Tools for the Support Professional

Microsoft® Press

MICROSOFT® PROFESSIONAL EDITIONS

Microsoft® Windows NT® Server

Resource Guide

Microsoft® Press

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

Library of Congress Cataloging-in-Publication Data
Microsoft Windows NT Server resource kit : for Windows NT Server
version 4.0 / Microsoft Corporation.
p. cm.
Includes index.
ISBN 1-57231-344-7
1. Microsoft Windows NT. 2. Client/server computing.
I. Microsoft Corporation.
QA76.76.W56M5317 1996
005.7'1369--dc20 96-33396
CIP

Printed and bound in the United States of America.

3 4 5 6 7 8 9 Rand-T 1 0 9 8 7

Distributed to the book trade in Canada by Macmillan of Canada, a division of Canada Publishing Corporation.

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 (206) 936-7329.

Part Number: 097-0001648 (Server Resource Guide)

Acquisitions Editor: Casey Doyle
Project Editor: Stuart J. Stuple

This book is dedicated to creating self-sufficient system administrators!

Contributors to this book include the following:

Managing Editor
Sonia Marie Moore
Lead Technical Editor
Pamela Miller
Technical Editors
John Andrilla, J. Elise Ellinghausen, Kate Robinson, Megan Sheppard
Proofreader
Kimberly O'Neal
Writing Manager
Peggy Etchevers
Technical Writers
Janet Anderson, Janice Breyer, Peter Costantini, Chris Dragich, J. Elise Ellinghausen,
Peggy Etchevers, Sharon Kay
Software Program Managers:
Louis Kahn, Ryan Marshall
Software Developer
Martin Holladay
Production Manager
Karye Cattrell
Production Lead
Nikole Faith
Production Team
Jay French, Patrick Ngo, Cathy Pfarr, Keri Segna, Jeff Weaver, Todd White
Indexers:
Jane Dow, Ronnie Maier, Barbara Sherman
Lead Graphic Designer:
Chris Blanton
Design Support Team:
Johnni Cutler, Flora Goldthwaite, Amy Iffland, Casey McGahan, Wendy Salvatori,
Gabriel Varela, Sue Wyble, Jan Yeager
Technical Consultants:
Cliff Hudson, Eugene Baucom, Phillip Carver, Joseph Dadzie, Scott Field, Jerry Gallagher,
Joseph Holman, Terence Hosken, Eric Hough, John Jacobs, Richard Lerz, Tom Lindley,
Jim Livingston, Janice Marsh, Oshoma Momoh, Joseph Pagano, Jim Truchon, Tom White,
David Winkler, and numerous other hardworking Windows NT Developers,
Program Managers, and Product Support Specialists
Product Support Liaisons:
Roger Bruist, Todd Hafer
Technical Marketing Liaisons:
Ty Carlson, Robert Lilleness

Contents

Part I About Windows NT Server

Part II Reliability and Recoverability

Part III Windows NT Server Troubleshooting

Figures and Tables

Figures

Tables

Introduction

Welcome to the *Microsoft® Windows NT® Server Resource Kit: Windows NT Server Resource Guide.*

The *Microsoft Windows NT Server Resource Kit* for version 4.0 consists of three new volumes and a single compact disc (CD) containing utilities for both Windows NT Workstation and Windows NT Server. An online version of the new, comprehensive *Windows NT Workstation Resource Guide* is also available on the CD. Update books for the *Windows NT Server Resource Kit* will be released on a semi-annual basis. They will contain new information and major revisions of existing topics.

The *Windows NT Server Resource Guide* presents detailed information that is specific mainly to the Windows NT Server operating system, plus topics that are either new for version 4.0 or reflect issues that our Product Support people consider timely and important. This information is intended to be an in-depth, technical supplement to the printed and online documentation included as part of the Windows NT Server version 4.0 product. It does not replace that information as the source for learning how to use the product features and programs. It is also supplementary to the information presented in the *Windows NT Workstation Resource Guide*, which is available in online format on the CD.

This Introduction includes the following types of information you can use to get started:

- The first section outlines the contents of this book, so that you can quickly find pertinent technical details.
- The second section introduces the *Windows NT Server Resource Kit* CD.
- The third section describes the support policy for the *Windows NT Server Resource Kit.*

About the Windows NT Server Resource Guide

This book includes the following chapters.

Part I, About Windows NT Server

Chapter 1, "Deploying Windows NT Server," discusses special considerations for deploying Windows NT Server 4.0. In general, you can deploy Windows NT Server using the techniques described for deploying Windows NT Workstation in Part I, "Windows NT Workstation Deployment," of the *Windows NT Workstation Resource Guide.*

Chapter 2, "Printing," describes the Windows NT printing architecture and flow of control, including printing from all possible network clients. Descriptions of each printing component, including the spooler and print server services, are included, as well as a printing troubleshooting guide.

Part II, Reliability and Recoverability

Chapter 3, "Disk Management Basics," describes managing and understanding disks to provide you with a basic understanding of disk hardware and how Windows NT accesses disks. The chapter also describes the Windows NT startup process so that you can troubleshoot problems effectively and recover from them.

Chapter 4, "Planning a Reliable Configuration," covers topics such as disaster planning, backup strategies, and mass storage strategies.

Chapter 5, "Preparing for and Performing Recovery," describes what information you should back up and how to use those backups to recover from problems.

Chapter 6, "Troubleshooting Startup and Disk Problems," discusses what you can do to isolate startup problems, along with possible causes of and how to recover from disk problems.

Chapter 7, "Disk, File System, and Backup Utilities," describes each of the utilities mentioned in the previous chapters and how to use them.

Part III, Windows NT Server Troubleshooting

Chapter 8, "General Troubleshooting," identifies tools that are available in Windows NT to help you troubleshoot problems. The chapter also contains information about troubleshooting hardware problems and how to use the information in the Registry to determine why services are not working correctly.

Index to this *Windows NT Server Resource Guide.*

Glossary is available online in the *Windows NT Workstation Resource Guide.*

Resource Kit Compact Disc

The CD that accompanies the *Windows NT Server Resource Kit* contains utilities that apply to information in the *Windows NT Workstation Resource Guide,* the *Windows NT Server Resource Guide,* the *Windows NT Server Networking Guide*, and the *Windows NT Server Internet Guide*. This new CD replaces all previous ones. It includes a collection of information resources, tools, and utilities that can make networking and working with the Windows NT platform even easier.

Note The utilities on this CD are designed and tested for the U.S. version of Windows NT version 4.0. Use of these utilities on any other version of Windows NT may cause unpredictable results.

A large Help file with explanations and user actions for the majority of the messages included in Windows NT version 4.0, and a large Help file of Performance Counter Definitions are just two of the major items included on the *Windows NT Server Resource Kit* CD. Updates to these files and others will be provided, when available, on the Microsoft Internet web site for the Windows NT Resource Kits. See the Rktools.hlp file for the exact site address, as well as the addresses of other Microsoft information sites.

After installing the *Windows NT Server Resource Kit*, please refer first to the following three files:

- The Readme.wri file, which contains a complete list of all the tools and utilities on the *Windows NT Server Resource Kit* CD and additional setup instructions for some of them.
- Either the Rkdocw.hlp (for Windows NT Workstation) or the Rkdocs.hlp (for Windows NT Server) file, which provides a single entry point for all of the major components of the Resource Kit's online documentation.
- The Rktools.hlp file, which provides an overview of the Resource Kit tools and utilities and basic instructions on how to use many of them, along with links to additional documentation and, in some cases, to the actual program files.

The most current corrections to those tools and utilities and their documentation, as well as the POSIX and Perl source code files, are available on the Internet at the following Microsoft FTP site:

ftp://ftp.microsoft.com/bussys/winnt/winnt-public/reskit/nt40/

The *Windows NT Server Resource Kit* CD includes a wide variety of tools and utilities to help you work more efficiently with both Windows NT Workstation and Windows NT Server. The following notes describe some of the enhancements made to the existing tools and utilities and introduce new ones that have been added for this version 4.0 release.

Batch Tools

- KiXtart 95, Kix32.exe, is a logon script processor and/or enhanced batch language for Windows NT and Windows 95 workstations in a Windows Networking environment. You can use it to do such things as display user information, connect to network resources, read/edit the Registry, and add/change program groups.

Computer Administration/Configuration Tools

- SrvInstW is a wizard for installing and deleting services and device drivers. This is a GUI form of the Instsrv.exe application that now also has increased functionality for detailing service parameters.

Computer and Network Setup Tools

- SysDiff.exe enables you to pre-install applications, including those that do not support scripted installation, as part of an automated setup.

Computer Diagnostic Tools

- Remote Kill is a service with both GUI and command-line clients. It enables a user to enumerate and kill a process on a remote computer. It also combines some of the functionality of Tlist.exe and Kill.exe.
- ShowAcls is a utility that enumerates the access rights for files, folders, and trees. It also allows masking to enumerate only specific ACLs.

Desktop Tools

- Desktop Themes from the Windows 95 Plus Pack are included now along with several new Windows NT-specific themes. These themes include a variety of visual, sound, and symbolic components that can enhance the look and feel of your Windows NT 4.0 desktop. Each desktop theme includes a background wallpaper, a screen saver, a color scheme, and a set of sounds, cursors, icons, and fonts.
- MultiDesk, Multidsk.exe, is a desktop switching program for Windows NT 4.0. It customizes the desktop wallpaper and colors, and separates executing programs into new desk spaces.

Disk/Fault Tolerance Tools

- DiskProbe, Dskprobe.exe, is a sector editor for Windows NT Workstation and Windows NT Server that enables a user with local Administrator rights to edit, save, and copy data directly on the physical hard drive that is not accessible in any other way.
- DiskMap produces a detailed report on the configuration of the hard disk that you specify. It provides information from the Registry about disk characteristics and geometry, the Master Boot Record, and the Partition Table for all the partitions on the disk.

File Tools

- ForFiles is a utility that enables batch processing of files in a folder or tree. You can run a command or give arguments to certain files. For example, you can run the **type** command on all files in a tree with the *.txt extension. Or you could execute every batch file (*.bat) on the C:\ drive with the "myinput.txt" filename as the first argument.
- LogTime is a utility that logs the date and time stamp for batch file calls.
- SetX is a utility that sets User vs. System environment variables.

Internet and TCP/IP Services/Tools

- Web Based Administration of Windows NT Server, WebAdmin, is an ISAPI DLL that will enable remote administration of Windows NT Server by means of shipping HTML browsers, such as Internet Explorer 2.0 and above, from Windows®, Macintosh, and UNIX platforms. Its purpose is to assist administrators when they do not have access to their traditional tools, such as when they are away from their normal workspace or "roaming."
- dbWeb is a gateway between Microsoft Open Database Connectivity (ODBC) data sources and the Internet Information Server (IIS). You can use dbWeb to publish data from an ODBC data source and provide familiar World Wide Web (WWW) hypertext navigation. While allowing users to create queries, dbWeb enables you to filter the data and sources that users can access and display.
- The Beta version of our Telnet Server, Telnetd.exe. You can use it to run command-line utilities, scripts, and batch files from operating system-independent clients. It is not intended to be a full, commercial-grade Telnet solution.

Network Diagnostic Tools

- SrvInfo.exe is a command line utility that displays local or remote computer-configuration information, including such things as whether it is a Server or a Workstation, the version, domain, PDC, IP address, CPU type and revision, drives, file systems, available drive space, mappings, services and status, network cards and bindings, and system up time.
- Updated and additional MIBs are included.

Network/Server Administration Tools

- Group Copy, GRPCPY, is a utility that enables you to copy the usernames in an existing group to another group that is in the same or another domain or a Windows NT computer. You must have at least Account Operator privileges to the domains with which you want to work.
- Remote Console is a client/server application that enables you to run a command-line session remotely, within which you may launch any other application. Few tools besides Remote Console run applications remotely. REMOTE is not a service, but simply redirects standard input and output, so console programs that take control of the video memory and the keyboard do not work with REMOTE. Along with RCMD and RSH, Remote Console offers you a new way to run consoles remotely and to take control of a CMD session.

Registry Tools

- The Registry Entries Help file, Regentry.hlp, has been updated again for this newest version. The chapter in the *Windows NT Workstation Resource Guide* that corresponded to this Help file has been discontinued.
- Over 10 new command-line Registry programs have also been added for searching, retrieving, and replacing keys in both local and remote Registries.

Tools for Developers

- PULIST is a utility that enumerates system processes and dumps process owners and IDs. You can use it against local or remote computers.
- Api Monitor, APIMON, is a utility that enables you to monitor the API calls that a process is making. This utility incorporates the functionality of the Application Profiler that will no longer ship with the *Windows NT Workstation Resource Kit.*
- CpuStress is a utility that loads down the processor, which is useful for evaluating how a system performs under heavy usage.
- Heap Monitor, HEAPMON, is a utility that enables you to view system heap information.

User Account Administration Tools

- Delete Profile, DELPROF, is a utility that deletes user profiles on Windows NT.
- SU now includes a GUI interface along with other major additions. SU enables you to start a process that is running as an arbitrary user.
- AddUsers has been updated to include such things as home-directory creation, global and local group deletion, and the ability to set the parsing token for the input/output (I/O) file.

Resource Kit Support Policy

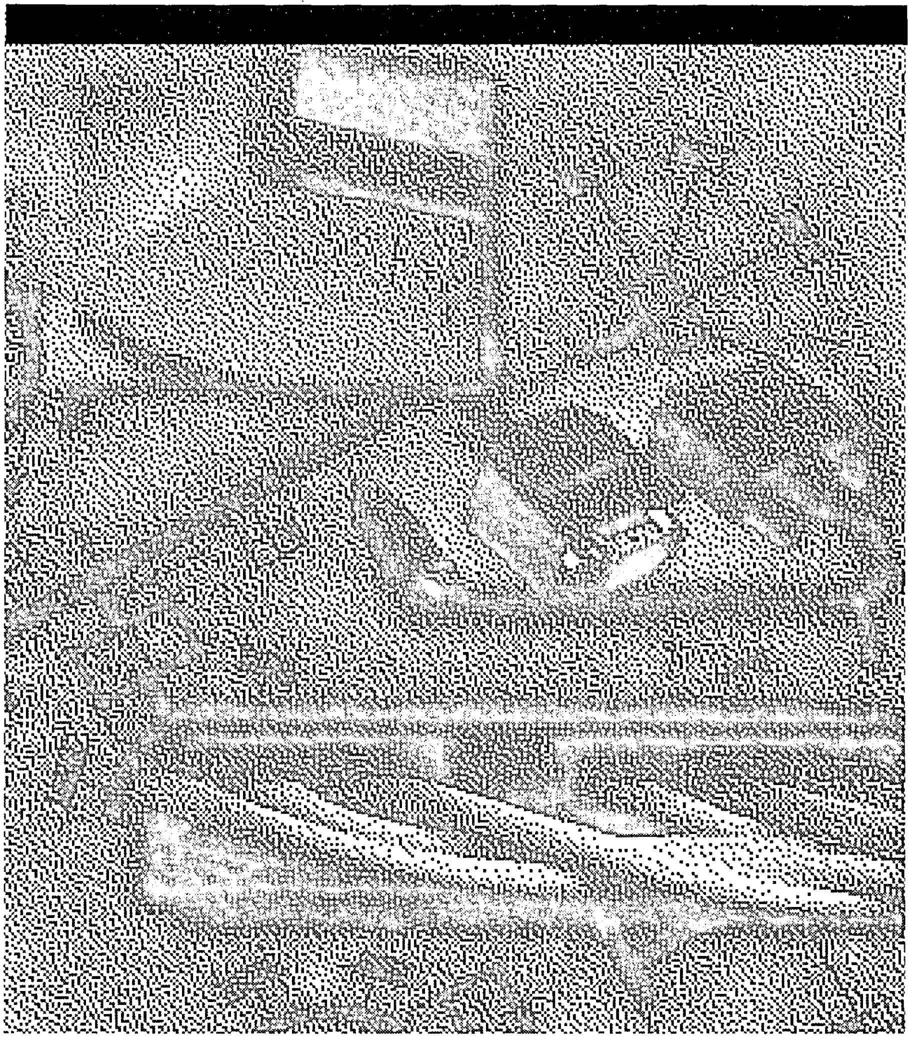
The SOFTWARE supplied in the *Windows NT Server Resource Kit* is not officially supported. Microsoft does not guarantee the performance of the *Window NT Server Resource Kit* tools, response times for answering questions, or bug fixes to the tools. However, we do provide a way for customers who purchase the *Windows NT Server Resource Kit* to report bugs and receive possible fixes for their issues. You can do this by either sending Internet mail to RKINPUT@MICROSOFT.COM or by referring to one of the options listed in the *Start Here* book, which is included with your Windows NT Server product. This mail address is only for *Windows NT Server Resource Kit* related issues.

The SOFTWARE (including instructions for its use and all printed and online documentation) is provided "AS IS" without warranty of any kind. Microsoft further disclaims all implied warranties, including, without limitation, any implied warranties of merchantability or of fitness for a particular purpose. The entire risk arising out of the use or performance of the SOFTWARE and documentation remains with you.

In no event shall Microsoft, its authors, or anyone else involved in the creation, production, or delivery of the SOFTWARE be liable for any damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use of or inability to use the SOFTWARE or documentation, even if Microsoft has been advised of the possibility of such damages.

PART I

About Windows NT Server



CHAPTER 1

Deploying Windows NT Server

Windows NT Server can be deployed to a large number of computers in the same way that Windows NT Workstation is deployed. The tools and techniques for doing so are described in Part 1, "About Windows NT Server" and Appendix A, "Answer Files and UDFs" of the *Windows NT Workstation 4.0 Resource Guide.*

There are some issues to be considered when deploying Windows NT Server that are not a factor in deploying Windows NT Workstation. These include which licensing mode to use, whether the server is to be a domain controller, and server-specific services that you might want to install.

Answer files and Uniqueness Database Files (UDFs) provide most of the tools you need for specifying options while deploying Windows NT Server. These are described in Chapter 2, "Customizing Setup" and Appendix A, "Answer Files and UDFs" of the *Windows NT Workstation 4.0 Resource Guide.* In particular, see the end of Appendix A for answer file entries that apply only to Windows NT Server.

Choosing a Licensing Mode

When installing Windows NT Server, you must specify whether the server is licensed in "per seat" or "per server" mode. You can provide this information in an answer file, as described in Appendix A, "Answer Files and UDFs" of the *Windows NT Workstation 4.0 Resource Guide.* For the installation to be completely unattended, you must supply this information in the answer file.

Installing a Domain Controller

When installing Windows NT Server, you can specify that the server be a member of a Workgroup or a Domain, or you can set the computer to be a domain controller. For the installation to be completely unattended, you must supply this information in the answer file.

The answer file entries for adding a server to a Workgroup or Domain are the same as those for workstations. To set the computer to be a domain controller, add the entry described under "Installing a Domain Controller" and "Setting the Server Role" in Appendix A, "Answer Files and UDFs" of the *Windows NT Workstation 4.0 Resource Guide.*

Replacing a Primary Domain Controller (PDC)

If the primary domain controller (PDC) is not available, and you want to configure a new PDC rather than promoting an existing backup domain controller (BDC), a new security identifier (SID) will be created for the domain. If the old PDC is then brought back on-line, it will not be able to process logon requests because it does not know the domain SID. To use both servers on the domain, you will need to reinstall Windows NT on one of them, while the other is available as a PDC. For more information see Chapter 1, "Managing Windows NT Server Domains" of the *Windows NT Server Concepts and Planning Guide.*

Installing Specific Services

To install network services, you will need to create one or more sections in the answer file in which you list the service you want to install. You can give these section any name you want. The sections you create must be specified with a line in the [Network] section of the answer file. The syntax is as follows:

```
[Network]
InstallServices = services_list_section[, services_list_section] . . .
```

Then, in the section you create, list the services you want to install. Parameters are provided in a third section, which you also create and give any name you want, as in the following example:

```
[Network]
InstallServices = ServiceList

[ServiceList]
SNMP = SNMPParameters

[SNMPParameters]
    Accept_CommunityName = MicrosoftCommunity
    Send_Authentication = YES
    Any_Host = NO
    Limit_Host = HostSystem
.....
.....
.....
```

The value to the left of the equals sign in the section you create to list the services is the same as the key for the service in the Registry(SNMP in the [ServiceList] section, in this example). The values for the parameters are also the same as those used in the Registry. Registry entries are documented in Part 5, "Windows NT Registry," and in the online help.

The following table lists server-specific services and the key used to specify them:

Service Name	Key
Alerter	AlertNames
Browser	Browser
DHCP Server	DHCPServer
FTP Server	FTPsvc
Internet Server	InstallInternetServer
NetBIOS over TCP/IP	NetBt
NetLogon	NetLogon
Remote Access (RAS)	RemoteAccess
Server	LanmanServer
WINS	WINS

Fault Tolerance Issues

If you are installing Windows NT on a mirrored partition, you must disable mirroring before deployment, and then reestablish mirroring after installation is complete.

Mirrored partitions are a feature you may have implemented in an existing installation of Windows NT Server, in which a copy of a partition on one disk drive is maintained on another disk drive. If the primary disk fails, no data is lost, since all information written to the primary disk was also written to the secondary disk.

CHAPTER 2

Printing

By using Windows NT Server as your network print server, you have a seamless way to print, no matter what operating systems your networked computers use. If your workstation computers are running Windows NT 4.0, network printing is as easy as “point and print.” For Windows NT and Windows 95 workstation computers, you don’t need to install a printer driver manually on the workstation computer before printing with Windows NT. Print resources seem to be provided automatically from each application.

Note Although you can certainly print locally from a Windows NT Server print server, this chapter concerns itself primarily with remote printing and regards the Windows NT Server as a dedicated print server. For information about printing locally from a print server, see the *Windows NT 4.0 Workstation Online Resource Guide.*

This chapter begins with a glossary of terms used throughout the rest of the chapter. The glossary is followed by an overview of the Windows NT printing process and the flow of control. After you have a clear picture of the printing process and are comfortable with the terms, you can move on to read in-depth descriptions of each component of the printing architecture based on the network protocols and operating systems of your networked computers. The chapter concludes with sections on print security, troubleshooting, and some common questions and answers.

For additional information about printing with Windows NT Server or Windows NT Workstation, refer to the following documentation.

Documentation Source	Descriptions
Microsoft Knowledge Base	An effective and consistent resource for troubleshooting printing problems.
Microsoft Technet	Microsoft White Papers, TechEd, and other articles are available on the Microsoft Technet CD.
Support Fundamentals for Windows NT, Chapter 13, "Printing with Windows NT"	A Microsoft Press self-paced training manual.

Glossary of Printing Terms

A *print device* is the actual hardware device that produces printed output—what you call a "printer" in casual conversation.

Print-device resolution is measured in *dots per inch* (DPI). The higher the DPI, the finer the resolution.

A *printer*, or *logical printer*, is the software interface between the operating system and print device. The printer determines how the document gets to the printing devices (for example, by means of a local port or to a remote print share) and to other parameters of the printing process. A single printer can send print jobs to multiple print devices, and multiple printers can send jobs to a single print device. For more detail about printers and print devices, see "Planning How Users Access Printers" in Microsoft Windows NT Server *Concepts and Planninge.* In the NetWare and OS/2 environments, this software interface is called a *queue.*

In Windows NT terminology, a *queue* is a group of documents waiting to be printed.

Print jobs are source code that contain both data and commands for print processing.

Print jobs are classified into *data types* based on what modifications, if any, the spooler must make to the job for it to print correctly. For example, one data type requires no modifications, whereas another data type requires the addition of a form feed to the end of the job.

The print *spooler* is a collection of dynamic-link libraries (DLLs) that receives, processes, schedules, and distributes print jobs.

Spooling is only one of the processes performed by the spooler. Spooling is the process of writing the contents of a print job to a file on disk. This file is called a *spool file*. In the event of power loss during printing, the spool file prevents loss of data so the print job can resume once power is restored. *Despooling* means reading the contents from a spool file and sending those contents to a print device.

Rendering means creating a print job. An application calls the graphics device interface (GDI). GDI takes the document information sent by the application, calls the printer driver associated with the target print device, and creates a print job in the printer language of the print device. The print device has "firmware" that interprets the submitted printer language and creates a bitmap for each page.

A *print server* is the computer that connects one or more print devices to the network and shares them with other networked computers. A print server can also be a special hardware device that connects a print device to the network with a net tap on one side and a parallel or serial port on the other side.

Although a Windows NT Workstation can function as a print server, and a Windows NT Server can function as a print client, this chapter considers the print server to be a Windows NT Server. For information about using a Windows NT Workstation as a peer-to-peer print server for a small LAN (10 or fewer connections), see the *Windows NT 4.0 Workstation Online Resource Guide.*

A *downlevel server* is a print server running a Windows product that is not Windows NT 4.0—for example, Windows for Workgroups or Windows NT 3.*x*.

Creating a printer means connecting to the print device, either over the network or over a serial or parallel port, naming the printer, and installing the printer driver. To create a printer, run the Add Printer wizard, and click the My Computer option. You'll be asked to install the printer driver and specify a port.

Connecting to a printer means connecting to the share on the computer that created the printer. To connect to a printer, run the Add Printer wizard, and click the Network printer option. You won't be required to install the printer driver. Only Windows NT and Windows 95 clients can "connect to" a printer shared out by a Windows NT 4.0 print server.

A *print client* is the computer that sends print jobs over the network to the print server. Print clients are sometimes also called client computers.

A *client application* is any application that creates a print job. The client can be an application on the print server or on a client computer on the network.

Network-interface printers are print devices connected directly to the network by means of their own network cards.

Print server services are software modules on the print server that receive print jobs and determine if the spooler should alter them. Different print server services support print jobs from various clients. For example, Services for Macintosh (SFMSRV) supports print jobs from Macintosh clients.

Printer drivers are software programs that enable applications to communicate fully and properly with print devices. Each print device can require unique codes and commnds to make available its special features, such as two-sided printing or custom paper sizes.

Anything described as *local* means that it exists on the same computer. For example, a print client seeks a printer driver locally (on the same computer) before downloading the driver from the print server.

Overview of the Printing Process

In the following overview of remote printing, Windows NT 4.0 Server is considered to be the print server. Some processes, or the software components that perform them, are slightly different for non-Windows NT print clients. For more details about different print clients, see "Print Clients" later in this chapter.

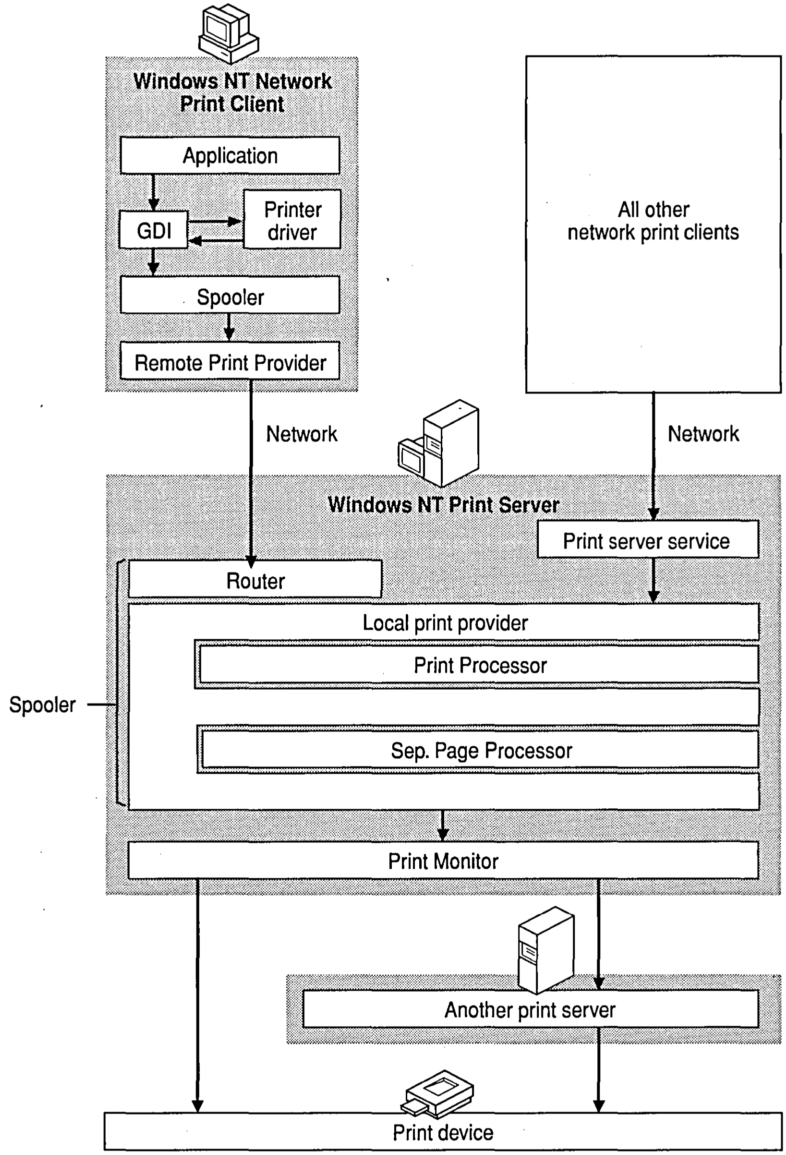


Figure 2.1 Windows NT Remote Printing Process

1. A user at a client computer chooses to print a document.

   If the client is running any Windows operating system and is printing from a Windows application, the application calls the graphics device interface (GDI). GDI calls the printer driver associated with the target print device. Using the document information from the application and the print device information from the printer driver, GDI renders the print job in the printer language of the print device. The application then calls the client side of the spooler (Winspool.drv).

   If the client is running a non-Windows operating system or is using a non-Windows application on a Windows operating system, another software component that functions similarly to GDI performs the same task.

2. The client computer delivers the print job to the print server.

   If the client is running Windows NT, the client side of the spooler (Winspool.drv) makes a remote procedure call (RPC) to the server side of the spooler (Spoolss.exe), which makes a direct application programming interface (API) call to the router (Spoolss.dll). The router polls the remote print providers, and the Windows remote print provider (Win32spl.dll) makes an RPC to Spoolss.exe on the print server, which receives the print job over the network.

   If the client is running another Windows operating system or is a Windows NT client who created a local printer port and redirected output to a network server (\\server\printer), the print job goes to the client computer's NetBIOS redirector. The redirector uses the protocol driver to send the job over the network to the print server's protocol, which delivers the job to the Windows NT Server service on the print server.

   If the client is a UNIX computer or any computer running LPR software, the print job goes to the TCP/IP protocol, which sends it over the network to the print server's LPD service.

   If the client is a Macintosh computer, the print job goes to AppleTalk, which sends it over the network to Services For Macintosh (SFMMON) on the print server.

3. The router or print server service receives the print job.

   On the print server, print jobs from Windows NT clients using Windows applications are enhanced metafiles (EMF). Some print server services for non-Windows NT clients assign a data type and others leave the data type blank. Print jobs with no data type assume the default data type in the **Print Processor** dialog box on the print server. For more information, see "Data Types" later in this chapter.

4. The router or print server service passes the print job to the local print provider on the server (a component of the spooler), which spools the print job (writes it to disk).

5. The local print provider polls the print processors. When a print processor recognizes the job's data type, that print processor receives the print job and alters it (or not) according to its data type to ensure that the job prints correctly.
6. Control of the print job passes to the separator page processor, which adds a separator page, if specified, to the front of the job.
7. The job is despooled to the print monitor.

   If the print device is bi-directional, the job first goes to a language monitor, which handles bi-directional communication with the printer and then passes the job to the port monitor.

   If the print device is not bi-directional, the job goes directly to the port monitor, which transmits the print job to the print device or to another server over a specific communications channel.
8. The print device receives the print job, translates each page into a bitmap, and prints it onto paper or another medium.

Print Jobs

Print jobs are not simply data: They are source code that contains both data and the commands for processing the data. The client application (with the graphics engine and the printer driver) creates print jobs. For example, Microsoft Word 7.0 combines data objects such as text, fonts, and graphics with information from a printer driver to create source code for a document—the print job.

Data Types

Because each type of print client (for example, UNIX clients and Windows NT clients) creates print jobs a little differently, a variety of print server services are required to receive and prepare the jobs. Some print server services assign print jobs a *data type*, which tells the spooler whether and how to modify the print job in order to print correctly. Some services leave the data type blank (in which case the default data type in the **Print Processor** dialog box on the print server is assigned).

If you create a printer, the default data type is EMF (if the printer is a PCL printer) or RAW (if the printer is Adobe PostScript). To change the default data type, use the following procedure:

▷ **To change the default data type for a printer you have created**

1. Click Start, and then click Settings.
2. Click Printers.
3. Click to select the printer for which you want to change the data type.
4. Click Properties on the **File** menu.
5. Click the General tab.
6. Click the **Print Processor** button.
7. In the Default Datatype list, click the data type.
8. Click **OK** in each dialog box.

The following sections describe the different data types.

EMF

Print jobs from Windows NT 4.0 clients are enhanced metafiles (EMF). Instead of the RAW printer data being generated by the printer driver, EMF information is generated by the Graphical Device Interface (GDI) before spooling. After the EMF is created, control is returned to the user, and the EMF is interpreted in the background on a 32-bit printing subsystem spooler thread and sent to the printer driver.

EMF files are more portable than RAW files. An EMF file can be printed on any print device, whereas a RAW file can be printed to only one print device model. In addition, the set of EMF files that represent all pages in a print job are typically smaller than a RAW file that contains the same print job.

The first portion of the print job's rendering is done on the client computer. The last portion is rendered on the print server. This is especially helpful when the print job is a very large file, because the client application is not tied up for the entire rendering time. This data type also ensures that fonts specified on the client computer are the same ones used by the print server.

RAW

The RAW data type tells the spooler not to alter the job at all—that the print job is ready to print as is. Most jobs from downlevel Windows clients are the RAW data type.

RAW [FF Appended]

This data type tells the spooler to assume the job is from an application that does not append a form-feed character (0x0C) to the end of each job. Without a trailing form-feed, the last page of the job does not print when sent to a PCL print device. The spooler appends a form-feed character to the end of the print job but makes no other alterations. None of the print server services supplied with Windows NT assign this data type, but you can set the default data type **Print Processor** dialog box to this type so that it affects jobs from Windows Network clients.

RAW [FF Auto]

This data type is similar to the RAW [FF Appended] data type, but RAW [FF Auto] tells the spooler to check for a form-feed character at the end of the job. It does not add a form feed if one is already present, and it makes no other alterations. None of the print server services supplied with Windows NT assign this data type, but you can set the default data type **Print Processor** dialog box to this type so that it affects jobs from Windows network clients.

TEXT

The TEXT data type tells the spooler that the job consists of ANSI text. The spooler uses the current print device driver to create a new print job that prints the text of the original job using the print device's factory default font, form, orientation, and resolution listed in the **Document Properties** dialog box. This is useful when the print job consists of simple text and the target print device (for example, a PostScript printer) cannot interpret simple text jobs.

Text files consist of numeric values from 0 to 255, where each value is mapped to a particular character or symbol. Several character-mapping schemes (character sets) are in common use, and text files contain no indication of which character set to use when displaying or printing the file. The TEXT data type assumes the ANSI character set, so it might print some characters incorrectly if the application that created the job does not use the ANSI character set. Most character sets are identical for the values 0 through 127, so this problem usually affects extended characters (those with values from 128 through 255).

The following table shows five examples in which common character sets use different numbers to represent the same character. The PC-850 character set is commonly used by MS-DOS®-based applications in Europe; ANSI is used by Windows-based applications; PC-437 is commonly used by MS-DOS-based applications in the United States; Roman-8 is the default PCL character set.

Character	PC-850	ANSI	PC-437	Roman-8
Lowercase C Cedilla	135	231	135	181
Lowercase AE Diphthong	145	230	145	215
Lowercase N Tilde	164	241	164	183
Lowercase Eth	208	240	-	228
Lowercase Eszet	225	223	225	222

PSCRIPT1

This data type indicates that the print job consists of PostScript code from a Macintosh client and that the target printer is a non-PostScript print device. The spooler interprets the PostScript code and creates a bitmap of the page. GDI32 and the printer driver can convert the bitmap into the language of the target print device.

Print Devices

Print devices are the machines we call "printers." They produce hardcopy output by interpreting the printer language and creating a bitmap for each page.

Print devices have their own input and output channels; jobs can be directed to input channels such as parallel or serial cables or network adapters. Output can be produced on different media, including paper, film, or fabric.

Print devices have their own internal processors which can be proprietary or general-purpose (such as the Motorola 680xx-series chips in Apple LaserWriter devices). Incoming data is stored in the print-device RAM, which can range from a few bytes to a hundred megabytes. Each print device has software stored in ROM, called "firmware," which interprets the programming language of a print job's source code. Examples of programming languages are PostScript, PCL, and HP-GL/2.

Printer Drivers

Printer drivers are the software that enables an application to communicate with the variety of available print devices, regardless of the device type, model, or programming language interpreter.

In general, printer drivers are composed of three files that work together as a printer-driver unit:

- A *printer graphics driver*. This file renders device driver interface (DDI) commands from the graphics engine to commands that a print device can understand. Each graphics driver renders a different printer language. For example, a Pscript.dll renders PostScript printer language.
- A *printer interface driver*. This file provides the user interface you see when you configure a printer.
- A *characterization data file*. This file provides information about the configuration capabilities of a specific make and model of print device, including what resolutions it is capable of, whether it can print on both sides of the page, and what paper sizes it can accept.

These three files work as a unit. For example, when you set up and configure a new printer, the interface driver queries the characterization data file and then displays the available choices. Then, when you print, the graphics driver queries the interface driver about your selections so that it can create the proper printer commands.

Printer drivers are generally not binary-compatible across hardware-processor platforms. Consequently, printer drivers must be installed on the Windows NT print server for each Windows NT client hardware platform. For example, when *x*86-based clients running Windows NT Workstation are served for printing by a Alpha AXP-based Windows NT Server, you must install x86-based printer drivers on the Alpha AXP-based print server.

The Windows NT and Windows 95 print client first attempts to use a local printer driver. If none is available, the Windows NT print server downloads the printer driver to the client computer's hard disk. The client application then uses its copy to create the print job.

In previous versions of Windows NT, printer drivers ran in user mode. With Windows NT 4.0, printer drivers run in kernel mode, thus reducing memory requirements and improving performance. For more information about user and kernel modes, see Chapter 5 "Windows NT Workstation Architecture," in the *Windows NT Workstation Resource Guide.*

Note Print clients running Windows NT 4.0 won't be able to use printer drivers downloaded from a print server running an earlier version of Windows NT. You'll need to make sure that the client computer has the 4.0 printer drivers, or better yet, use 4.0 on your print servers. A print server running Windows NT 4.0 can supply earlier version printer drivers to downlevel Windows print clients, if necessary.

Platform-specific printer drivers are available for the following:

- Intel x86-based computers running Windows NT 4.0, 3.5, or 3.51 and Windows 95
- MIPS RISC-based computers running Windows NT 4.0, 3.5, or 3.51 and Windows 95
- Digital Alpha AXP-based computers running Windows NT 4.0, 3.5, or 3.51 and Windows 95
- IBM PowerPC running Windows NT 4.0, or 3.51 and Windows 95

Print devices can be classified as one of three types: raster, PostScript, or plotter. To support these three classes of print devices, Windows NT provides the following generic printer drivers:

- Universal printer driver
- PostScript printer driver
- HP-GL/2 plotter driver

Universal Printer Driver

The Universal printer driver (Unidriver) is sometimes referred to as the *raster driver* because it supports raster graphics printing. The Unidriver can carry out requests on most types of printers. Each print device vendor provides a device-specific minidriver (called a *characterization file*) that works with the Unidriver to communicate with its print devices.

This driver includes support for 24-bit color, scaleable TrueType fonts, device fonts, compression-run length encoding (RLE), and Tag Image File Format (TIFF) version 4.0. It also includes mechanisms that provide for smaller, more efficient bitmaps. These mechanisms include ignoring whitespace and supporting rules. (Hewlett-Packard LaserJet and compatible print devices use rules to print repeating elements on a page—such as bullets in a list—by repeating the information from a single bit of source code.)

The Universal printer driver consists of the following component files:

- Rasdd.dll is the printer graphics driver for printer languages based on raster (bitmap) images, including PCL and most dot-matrix printer languages.
- Rasddui.dll is the printer interface driver.
- *Raster minidriver* is the characterization data file. The filename for this component depends on the print device or printer family. Check the documentation that accompanies your print device.

The Windows NT Universal driver is a generic, text only (TTY) driver. It prints only text, and it prints it in the native font of the print device, regardless of any formatting in the original document.

PostScript Printer Driver

The Windows NT PostScript driver uses Adobe version 4.2-compatible PostScript printer description (.ppd) files. (Windows NT does not use the .wpd or .mpd files used by Windows 3.1). This driver supports key features, including binary transfer compression from Level II, resolution, and paper source.

The component files of the Windows NT PostScript printer driver are as follows:

- Pscript.dll is the printer graphics driver.
- Pscriptui.dll is the printer interface driver.
- *x*.ppd is the PostScript printer description (PPD) characterization data file.

Note PPD files are the only printer driver files that are generally binary-compatible across processors and platforms.

HP-GL/2 Plotter Printer Driver

The Windows NT plotter driver supports a variety of plotters that use the HP-GL/2 language. Windows NT does not support HP-GL. The output from the Windows NT plotter driver requires a plotting device that can process all of the enhancements built into the HP-GL/2 language.

The Windows NT plotter driver consists of the following component files.

- Plotter.dll is the printer graphics driver.
- Plotui.dll is the printer interface driver.
- *x*.pcd is the characterization data file.

Establishing Printers

With Windows NT 4.0 you can establish printers with the Add Printer wizard or by using Point and Print.

Windows NT 4.0 provides the Printers folder to manage printing. Within it, the Add Printer wizard simplifies the task of establishing printers You can open the Printers folder in the following ways:

- Click Start, point to Settings, and then click Printers.
- In My Computer, double-click the Printers folder.
- In Control Panel, double-click the Printers folder.

▷ To establish a printer with the Add Printer wizard

- In the Printers folder, double-click the Add Printer icon.

  The Add Printer wizard leads you through the process of setting up and configuring a printer. The only differences between establishing a network printer and a local printer are as follows:

  - If you are establishing a local printer, you must install a printer driver. This action "creates" the printer on your computer. You can share the printer with other network computers.
  - If you are establishing a remote printer, you must specify the path to the network printer or browse to find its network location. This action "connects to" the printer on the print server.

▷ To install a printer using Point and Print

1. Open the Printers folder.
2. Double-click **Network Neighborhood.**
3. Click the print server.
4. Drag the printer icon to your Printers folder.

   This action "connects to" the printer on the print server.

Creating vs. Connecting to a Printer

The method used to establish a printer on a client computer dictates the configuration of the client computer in relation to the printer. The following table details the various methods.

Connection Method	Windows NT Sending Software	Required Network Protocols	Windows NT Receiving Software
Connect to network printer in Add Printer wizard	Spoolss.exe	TCP/IP, NetBEUI, NWLink	Spoolss.dll
Create local printer in Add Printer wizard and Print To options for a UNC device	Windows NT Workstation Service (NetBIOS redirector)	TCP/IP, NetBEUI, NWLink	Windows NT Server service
Create local printer and Print To options for an LPR port	LPR (Lpr.exe) Port Print Monitor	TCP/IP	TCP/IP (LPD) Print Service
Create local printer using Chooser and Print To options for an AppleTalk device	AppleTalk Print Monitor	AppleTalk	Services For Macintosh

Notice that only Windows NT and Windows 95 client computers can actually "connect to" a network printer served by a Windows NT print server. When a Windows NT or Windows 95 print client initiates a print request, the required printer driver is downloaded from the Windows NT print server if it is not already on the client's hard disk. All other client computers must "create" the printer—that is, they must install the printer driver directly on their hard disks, specify a port, name the printer, and so on.

Windows NT and Windows 95 print clients can also create a remote printer served by a Windows NT print server. Each method has advantages and disadvantages. Connecting to a remote printer is easier and faster than creating one. If the Windows NT client has connected to a printer, the print job doesn't spool on the client machine, so no spool options are available. (Windows 95 clients always spool locally and again remotely.) The "connected" client also cannot queue print jobs locally. Creating a printer gives the user more control, but that control is not always needed.

Specifying Virtual Printer Memory

You can change the amount of virtual memory that your PostScript printer has available for storing fonts. The PostScript driver uses a default setting recommended by the printer manufacturer for virtual memory.

To determine the right value, copy the Testps.txt file (supplied with the Windows NT Resource Kit) to the printer, and use the recommended virtual memory value printed on the resulting page.

▷ **To change your PostScript printer's virtual memory**

1. Click Start, point to Settings, and then click Printers.
2. Click to select the printer.
3. Click **Properties** on the **File** menu.
4. Click the Device Settings tab.
5. Double-click Available PostScript memory.
6. Type the value from Testps.txt in the Change Settings box.
7. Click **OK.**

Print Clients

Print clients send print jobs to a Windows NT print server. The print server services or the spooler receive the print job as an input/output (IO) request that must be directed to the appropriate system process.

Each print server service uses different logic to determine how the job should be altered. The following sections describe protocols, general architecture, and process logic for each of the following types of print client:

- Windows NT network clients
- Windows NT local clients

  For information about printing locally see the *Windows NT Workstation Online Resource Guide.*
- Windows 95 network clients
- 16-bit Windows network clients
- MS-DOS network clients
- UNIX clients
- NetWare clients
- Macintosh clients

All Windows print clients receive printer alerts confirming successful prints or indicating errors. To disable printer alerts on a specific machine, add the following entry to that computer's [Network] section of the System.ini file:

```
SendPrintMessages=no
```

Note Disabling WinPopup does not disable printer alerts.

Windows NT Network Clients

Windows NT network clients can use any of the following network protocols to send print jobs to a Windows NT Server print server:

- NetBEUI
- NWLink
- TCP/IP
- AppleTalk

If the Windows NT client computer connects to a printer, it sends print jobs to the server by making remote procedure calls from the client's print router to the server's print router. If the client created the printer, the print server service handles delivery of the print job.

Windows NT print clients now use enhanced metafile (EMF) data type instead of RAW. EMF allows the print job to be spooled to the server and rendered at the server into the appropriate print format. Remote rendering significantly speeds up the client computer's return to task.

Windows NT network clients need not worry about printer drivers, because if the client doesn't have a needed driver, it is automatically installed on the client's hard disk from the Windows NT server when the client begins its print operation. If, however, the client computer is running Windows NT 4.0 and the server is running an earlier version, the printer drivers won't work. A Windows NT 4.0 print server can supply earlier version of printer drivers where necessary, but a Windows NT 4.0 client cannot use earlier versions of printer driver.

Windows 95 Network Clients

Windows 95 print clients send print jobs to a Windows NT print server by using one of the following MS Network client redirectors:

- LAN Manager
- Microsoft Network Client 3.0 for Windows

The MS-Network redirectors send jobs by using the following protocols:

- NetBEUI
- NWLink
- TCP/IP

Windows 95 generates EMF data type, which always spools locally. Spool files are played back locally to generate RAW data type before being sent to the print server. The print server spools the RAW file again.

The Windows NT print server receives print jobs from Windows 95 clients through the Windows NT Server service.

16-bit Windows Network Clients

There is little difference between the different versions of the 16-bit Windows platforms. They all support printing from MS-DOS-based applications and from Windows-based applications by using the 16-bit printer driver installed on the computer. They typically send jobs to a Windows NT print server by using one of the following MS Network client redirectors:

- LAN Manager
- Microsoft Network Client 3.0 for Windows
- Windows for Workgroups built-in redirector

The MS-Network redirectors send jobs by using the following protocols:

- NetBEUI
- NWLink
- TCP/IP

The Windows NT print server receives the job through the Windows NT Server service, which typically does not alter the print job. 16-bit Windows clients might also run third-party software for sending jobs to other print servers, such as LPR software for sending print jobs to UNIX print servers. This software is often able to send jobs to Windows NT as well.

MS-DOS Network Clients

MS-DOS clients usually use an MS Network client redirector to send print jobs to Windows NT print servers. They typically use the NetBEUI or TCP/IP protocol, but they might also print with third-party software (such as an LPR utility). Some MS-DOS-based applications create print jobs that contain only ASCII text. Unlike Windows clients that share the same driver for a particular print device, those MS-DOS-based client applications contain an internal version of the driver for each specific print device. This internal driver might not be the correct version of the driver. Often you can modify the driver and printer settings within the MS-DOS-based application or by using an accompanying utility. Check the documentation for the application or utility.

MS-DOS-based applications might not be *network-aware*, meaning that the application does not allow for a network redirector to forward the print job over the network to a print server. Some network-unaware applications try to print by directly dictating to the parallel port hardware, or they might rely on MS-DOS to control printing and be unable to print correctly to a network print server. Other network-unaware applications assume that they have exclusive control of the printer and therefore don't properly terminate their print jobs. As a result, nothing else prints until the application terminates. You might need to consider upgrading to a network-aware application in order to print over the network.

UNIX Clients

Windows NT supports UNIX printing clients by providing the TCP/IP Print Server (LPD) service. This service can receive print jobs from UNIX systems or other operating systems—including Windows NT—that have LPR client software. The client software must support RFC 1179. This is problematic, because many systems do not support this specification. Others that claim to support it do not implement it entirely, or they supply private extensions to the specification that their users have come to rely on but that don't exist except on that system. For more information about how to correct problems printing from systems that don't comply with RFC 1179, see "Line Printer Monitor (Lpdmon.dll)" later in this chapter.

The LPR protocol does not pass detailed error status information back to the LPR client. If anything goes wrong, from severe problems (such as the server being too busy to process requests) to print device problems (such as running out of paper), the LPR protocol reports the same error condition.

NetWare Clients

NetWare clients use IPX/SPX-compatible transport to send print jobs over the network to the Windows NT print server. The server must have installed Microsoft File and Print Services for NetWare. This utility, which is a separate product from Windows NT, enables the NetWare client to see the Windows NT printers as print queues.

Macintosh Clients

Macintosh clients send jobs over AppleTalk to a Windows NT print server. To the Macintosh client, the Windows NT computer looks like another AppleTalk device in a zone—either an AppleShare print server or a standalone print device. On the Windows NT Server computer, the SFMSRV receives the jobs. For more information about SFMSRV see, "Services for Macintosh Service" later in this chapter, and "Services for Macintosh Print Processor (SFMPSPRT)" and "Macintosh Port Monitor (SFMMON)" later in this chapter.

Print Server Services

Print server services are the software on the Windows NT print server that receive print jobs from specific types of network clients, determine whether the spooler should alter them, and then send them to the spooler by calling the spooler application program interface (API).

Each print server service uses different programmatic logic and makes different assumptions about print jobs from its supported print clients. These differences are detailed in the sections describing each service.

Windows NT Server provides the following print server services:

- Windows NT Server Service
- TCP/IP Print Server Service—known as Line Printer Daemon (LPD)
- Services For Macintosh (SFMSRV)

In addition, other print services may be installed, such as:

- Microsoft File and Print Services for NetWare (FPNW)

The following table lists each print server service and the type of print client processed by the service.

Service	Registry Name	Supported Clients
Windows NT Server service	Server	MS Network clients using NetBIOS redirectors on MS-DOS, Windows, Windows for Workgroups, or Windows NT computers
Services for Macintosh	Sfm	AppleTalk clients on Macintosh computers
TCP/IP Print service	Lpdsvc	LPR (line printer) clients on UNIX or Windows NT computers where the TCP/IP protocol is used
File and Print Services for NetWare	Frnw	Computers running NetWare client software where IPX/SPX compatible transport protocol is used

Windows NT Server Service

The *Windows NT Server service* receives jobs from applications and computers that are using the MS Network client and NetBIOS redirectors. This includes print clients using any of the following:

- Windows NT

  The Windows NT Server service receives jobs from Windows NT clients only if the client has created the printer in his or her own machine. If the client has connected to the printer, the print job goes directly to the server's spooler. For more information about creating versus connecting to a printer, see "Creating vs. Connecting to a Printer" earlier in this chapter.
- LAN Manager
- MS Network Client (often used on clients running MS-DOS alone or MS-DOS with Windows Versions 3.0, 3.1 or 3.11)
- Windows For Workgroups
- Windows 95

The Windows NT Server service does not set the data-type value when it submits the print job to the spooler. Instead, the spooler uses the default data type specified in the **Print Processor** dialog box on the print server. You can change the default data type. For more information, see "Data Types" earlier in this chapter.

Note If you select EMF or PSCRIPT1, the print spooler ignores this data and instead uses the RAW data type to process print jobs. Detailed information on print spooler processing of data types is provided later in this chapter in the "Print Processors" section.

TCP/IP Print Service (LPD)

The TCP/IP Print service is generally referred to as LPD, which stands for *line printer daemon* (or "service"). LPD receives print jobs from line printer remote (LPR) clients. LPR clients are often on UNIX systems, but LPR software exists for most operating systems, including Windows NT.

LPR is one of the network protocols in the TCP/IP protocol suite. It was originally developed as a standard for transmitting print jobs between computers running Berkeley UNIX. The LPR standard is published as Request For Comment (RFC) 1179. Previous versions of Windows NT supported TCP/IP printing as documented in RFC 1179, which describes an existing print server protocol widely used on the Internet for communicating between line printer daemons but does not specify an Internet standard. Different implementations support different options.

Windows NT 4.0 has added enhancements to support the most popular and requested options. Windows NT now suppots multiple data files per control file, and when used in "print through" mode as an intermediate spooler, it correctly passes the hostname parameter through the Windows printing subsystem. Windows NT 3.5*x* sent all TCP/IP print jobs from Windows NT computers from TCP ports 721 through 731. If enough jobs were sent in quick succession the ports could become a bottleneck causing a delay. For Windows NT 4.0 LPR print jobs are sourced from any available reserved port between 512 and 1023.

The LPR protocol does not pass detailed error status information back to the LPR client. If anything goes wrong, from severe problems (such as the server being too busy to process requests) to print-device problems (such as running out of paper), the LPR protocol reports the same error condition.

LPD receives jobs from LPR clients and submits them to the spooler. LPR clients always send a "control file" (actually, a data structure within the print job) containing administrative information with each print job. LPD assigns a data type based on the control commands in that control file.

If the client sends the **f**, **o**, or **p** control command, the LPD assigns the TEXT data type to the print job—which tells the spooler to edit the job to make sure it prints. If the client sends the **l** control command, the LPD assigns the RAW data type to the print job—which tells the spooler the print job needs no editing to print correctly. For more detail about data types, see "Data Types" earlier in this chapter.

Control commands are documented in the LPR specification, Request For Comment (RFC)1179, sections 7.17 through 7.29. For detailed information about RFC 1179, refer to the Windows NT Knowledge Base article, "Text of RFC1179 Standard for Windows NT TCP/IP Printing," reference number Q124734.

Note Notice that all the control commands defined in RFC 1179 are case sensitive. Notice also that many printer languages, including PCL, rely heavily on the ESC control character, which the **f** control command causes to be filtered from the print job. Do not use the **f** control command when sending print jobs that contain printer commands.

Because LPD assigns a data type explicitly, the default data type found in the **Print Processor** dialog box has no effect on print jobs received by LPD. To change the behavior of LPD, you must reconfigure the LPR client application to send a different control command with the print job. To reconfigure a LPR client application, consult the documentation for the application.

File and Print Services for NetWare

Microsoft File and Print Services for NetWare (FPNW) is a Windows NT Server utility that makes Windows NT Server look like a NetWare 3.*x*-compatible file and print server to computers running NetWare client software. Windows NT Server integrates into NetWare environments with IPX/SPX Compatible Transport.

Using FPNW, NetWare clients can print to printers attached to Windows NT print servers or to NetWare-compatible printers attached directly to the network. The Windows NT printers appear as print queues on a NetWare network. To create printers, the NetWare client should select LPT1 port for a printer attached to the Windows NT print server. For a network-attached printer, the clients selects either NetWare_Pserver_0 or NetWare_Pserver_1 port.

Services for Macintosh Service

The Services For Macintosh service (SFMSRV) receives print jobs from Macintosh clients. From the client's perspective, a printer shared through this service looks just like a share on an AppleShare server or a stand-alone AppleTalk print device.

The SFMSRV assigns the RAW data type to all jobs targeted to PostScript printers and also assigns the PSCRIPT1 data type to all jobs targeted for non-PostScript print devices. For additional information about processing of Macintosh print jobs, refer to the section "Services for Macintosh Print Processor (SFMPSPRT)" later in this chapter.

Print Spooler

The preceding sections described network print clients, print server services, print jobs, data types, and print devices. These are entities at the beginning and end of the printing process. The following sections describe the Windows NT print spooler components and the processing that takes place on a Windows NT print server after the print job is created and before the print job reaches the print device.

The following figure shows the main spooler components used to process jobs on a Windows NT print server. Each component uses the services of the component directly below it.

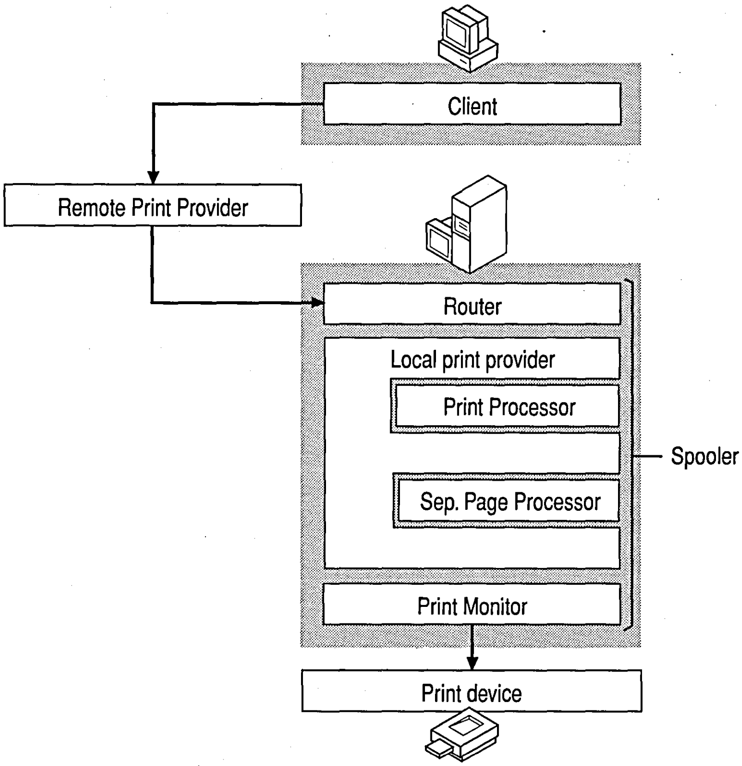


Figure 2.2 Print Spooler Components

As shown in the figure, the components below the client and above the print device are collectively called the *spooler*. The spooler is a series of 32-bit virtual device drivers and DLLs consolidated into a single architecture. It provides smooth background printing by using background thread processing. This means that the spooler passes data to the printer only when the printer is ready to receive more information. In Windows NT, the spooler components are implemented as a service that you can stop and restart from the Services icon on the Control Panel or from the command line by using the **Net Stop Spooler** and **Net Start Spooler** commands.

Important If you stop the spooler service, you will not be able to print until you start it again.

The spooler components reside on both the Windows NT print server and on the Windows NT client computer. Client computers running a different operating system might have different components, and they might interact differently with the Windows NT print server's spooler components. The following table lists the components of the Windows NT spooler:

Spooler Component	Description
Router	Winspool.drv, Spoolss.exe, Spoolss.dll.
Local Print Provider	Localspl.dll
Remote Print Providers	Win32sp.dll for Windows print servers.
	Nwprovau.dll for Netware print servers.
Print Processors	Winprint.dll
	Sfmpsprt.dll
Print Monitors	Localmon.dll
	Hpmon.dll
	Lprmon.dll
	Sfmmon.dll
	MARKVISION
	DECMON

Each spooler component is described in more detail in the following sections.

Router

When a Windows NT application has a print job it wants to send to the printer, it communicates with the client side of the spooler (Winspool.drv). This file makes an RPC to the server side of the spooler (Spoolss.exe), which makes a direct API call to the router (Spoolss.dll—which is also on the server side of the spooler).

The router passes control of the print job to the appropriate print provider. In the case of local printing, the local print provider takes the print job. If the print job is going to a server on the network, the router passes the print job to the appropriate remote print provider—the Windows remote print provider or the Novell NetWare print provider.

Do not confuse the software router component of the Windows NT spooler with the physical hardware router used in networks.

Remote Print Providers

When a client computer connects to a remote printer and sends a print job to it, the router polls each remote print provider on the client computer. The router passes control of the print job to the first remote print provider that recognizes the printer name.

Windows NT supplies the following remote print providers:

- Windows Network print provider (Win32spl.dll) transfers jobs to Windows Network print servers (such as print servers running Windows NT or Windows for Workgroups).
- Novell NetWare print provider (Nwprovau.dll) transfers jobs to Novell NetWare print servers.

Note Neither of these remote print providers performs spooling.

Windows Network Print Provider

If the local copy of the Windows network print provider, Win32spl.dll, recognizes the printer name, it performs additional processing based on the type of print server to which the job is going. If the print server is running Windows NT, Win32spl.dll makes remote procedure calls to the router (Spoolss.dll) on the print server. The router receives the print job over the network and passes it to the local print provider as if one of its own local clients had submitted it.

If the remote print server is a downlevel server, Win32spl.dll sends a message to the Windows NT NetBIOS redirector. The redirector forwards the job over the network to the downlevel server.

The functions provided by the Windows network print provider are illustrated in the following figure.

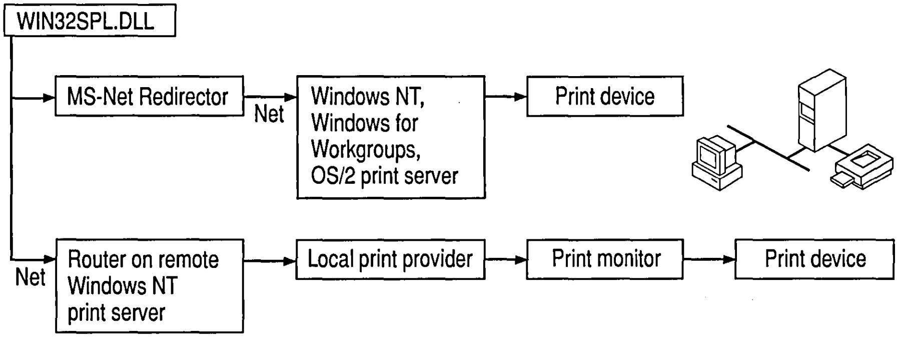

WIN32SPL.DLL
MS-Net Redirector
Net
Windows NT, Windows for Workgroups, OS/2 print server
Print device
Router on remote Windows NT print server
Net
Local print provider
Print monitor
Print device

Figure 2.3 The Windows Network Print Provider (Win32spl.dll)

Novell NetWare Remote Print Provider

If the NetWare print provider (Nwprovau.dll) recognizes the printer name when polled by the router, it takes control of the print job. The NetWare print provider sends a message to the NetWare workstation service (Nwwks.dll), which in turn passes control to the NetWare redirector. The NetWare redirector transmits the print job over the network to the NetWare print server.

Note To send print jobs over Windows NT to a NetWare print server, you must install Gateway Services for NetWare. To receive print jobs from NetWare print clients, you must install File and Print Services for NetWare.

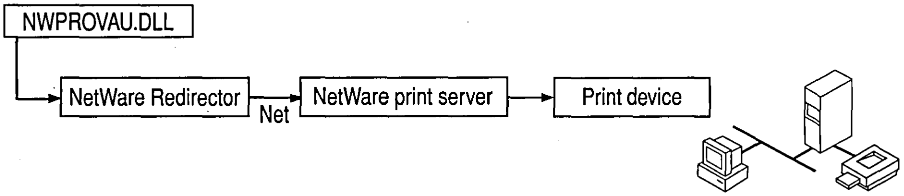


Figure 2.4 The NetWare Print Provider (Nwprovau.dll)

Local Print Provider (Localspl.dll)

The *local print provider* writes the print job contents to a spool (.spl) file and tracks administrative information (such as user name, document name, and data type) in a shadow (.shd) file. By default, both files are written to *%systemroot%*\SYSTEM32\SPOOL\PRINTERS.

If, however, the hard drive partition containing Windows NT Server doesn't have sufficient disk space to accommodate the spool files, you can change the location of spool files by changing the server properties.

▷ To change the location of spool files

1. Create a new spool file directory.
2. Click Start, and then point to Settings.
3. Click **Printers**.
4. Click **Server Properties** on the **File** menu.
5. Click the Advanced tab.
6. In the Spool Folder box, type the path for the new spool file directory.
7. Click OK.

Note You must create a directory for the new spool file location. If you attempt to spool directly to the root (C:\ or D:\, for example) the spool file will revert to the default spool directory.

Spooling a print job to a file protects the print job by saving it on disk. Should the print server suffer a power failure or other serious event before printing all jobs in the queue, the spool and shadow files on the server's hard disk preserve each job and prevent any loss of data once processing resumes.

By default, spool and shadow files are deleted after the job prints. However, you can enable spooler event logging to get valuable information about printer traffic, hard disk space, and other printing maintenance issues.

▷ To enable spooler event logging

1. In the Printers folder, click the printer to select it.
2. Click **Server Properties** on the **File** menu.
3. Click the Advanced tab.
4. Click Enable spooler event logging to add a check mark.
5. Click **OK**.

Next, LOCALSPL polls the installed print processors (such as Winprint.dll and Sfmpsprt.dll) to see if one of them recognizes the job's data type set. If the data type is not set, the Winprint.dll print processor receives the job and uses the default data-type set **Print Processor** dialog box on the print server.

You can set a new default location or override the default location on a printer-by-printer basis by manually editing the Registry. However, before doing so, read the "Spool File Security" section later in this chapter.

Enhanced Metafile (EMF) Spool Files

Enhanced metafiles (EMF) are one type of spool file used by the default Windows NT print spooler. EMF spool files reduce the time between the initiation of a print request and when control is returned back to the operating system. This is done by storing only the GDI function calls that produce the graphic object the application wants printed. The much more time-consuming execution of function calls can then be carried out later, in the background, when the spool file is "played back."

The way EMF spool files are encoded also provides the advantage of printer device-independence. In other words, a picture measuring 2 inches by 4 inches on a VGA display and stored in an EMF maintains those original dimensions whether it is printed on a 300 dpi laser printer or on a 75 dpi dot matrix printer.

RAW Spool Files

If the print job's data type is RAW, the spool file's data type is RAW. These files are device-dependent. The spooled data is destined and formatted for a particular device and does not need to be printable on a different device. An example of a RAW spool file is an encapsulated PostScript (EPS) file, which is formatted to be understood by the PostScript printer for which it is destined, but which is just RAW data to the Windows NT spooler.

Print Processors

A print processor works in conjunction with a printer driver to despool the spooled print jobs during print spool file playback.

Print processors are the components that make necessary alterations to print jobs, based on the data type of the print job. A print processor might recognize only one data type, or it might recognize several data types. Windows NT supplies the following print processors:

- Windows print processor (Winprint.dll)
- Macintosh print processor (Sfmpsprt.dll)

A print device vendor might develop a custom print processors if:

- The vendor develops its own printer driver.
- The vendor supports a data type different from those supported by the Microsoft print processor.

Additional print processors can be supplied by third-party software vendors to support custom data types.

Windows Print Processor (Winprint.dll)

This print processor supports five data types: EMF, RAW, RAW [FF Auto], RAW [FF Appended], and TEXT. For a detailed description of each data type, see "Data Types" earlier in this chapter.

Services for Macintosh Print Processor (Sfmpsprt.dll)

This print processor is available only with Windows NT Advanced Server version 3.1 or Windows NT Server version 3.5, 3.51, and 4.0. By default, this print processor (SFMPSPRT) is installed only when Services For Macintosh is installed on the Windows NT Server computer.

SFMPSPRT supports the PSCRIPT1 data type. This data type indicates that the job is Level 1 PostScript code from a Macintosh client, but the target printer is not a PostScript printer. The spooler sends the PostScript code through a Microsoft TrueImage® raster image processor (RIP), supplied with Services for Macintosh. The raster image processor creates a series of one-page, monochrome bitmaps at a maximum of 300 DPI. The Windows NT print spooler sends the rasterized images, or bitmaps, to the print driver for the target printer. The print driver returns a job that prints the bitmaps on the page.

Because the RIP bitmaps are monochrome (black and white) and not more than 300 DPI, the target printer driver produces final output that is monochrome and not more than 300 DPI, even if the target printer driver supports color or higher resolutions. The restrictions are in the RIP software itself, not in the Windows NT printer drivers.

Several third-party Win32® RIP packages are commercially available for Windows NT version 3.1 and above for those who need a higher-end RIP.

Print Monitors

Windows NT 4.0 supports two kinds of print monitors: *language monitors* and *port monitors*.

A language monitor is necessary to support bi-directional print devices. A bi-directional print device supports two-way communication between the print device and the spooler running on its print server. The language monitor allows the spooler to configure and monitor the status of a bi-directional printer. A language monitor can also add data (such as printer control information) to the print stream. When you create a printer, you install a printer driver. If a language monitor is associated with that printer driver, all data that flows from the printer driver to the print device goes through the language monitor before it goes through the port monitor and out to the printer.

Two-way communication between computer and print device allows a system administrator to configure the printer and to monitor printer status. The spooler running on the computer can request configuration and status information from the printer, and the printer can send unsolicited status information to the computer (such as the fact that the paper tray is empty). A common language is necessary for the two devices to understand each other.

The language monitor supplied with Windows NT 4.0 uses Printer Job Language (PJL). If a print-device vendor uses a printer language other than PJL, that vendor can develop a language monitor for it. A vendor might also develop a language monitor to add data, such as printer-specific control information, to the print stream going to a single directional printer.

Port monitors control the I/O port to which the physical print device is connected. The local print monitor executable (Localmon.dll) supplied by Microsoft controls parallel and serial I/O ports that may have a print device connected to them. Print devices connected to different types of I/O ports such as SCSI or Ethernet ports on a network card or adapter, are controlled by vendor-developed port monitors.

The print monitors supplied with Windows NT are described in the following sections.

- Local print monitor (Localmon.dll)
- Hewlett Packard Network Port print monitor (Hpmon.dll)
- Line Printer (LPR) Port print monitor (Lprmon.dll)
- Macintosh print monitor (Sfmmon.dll)
- Digital Network Port print monitor (Decpsmon.dll)
- LexMark Mark Vision print monitor (Lexmon.dll)
- PJL Language Monitor (Pjlmon.dll)

The Ports tab of the **Printer Properties** dialog box lists the default Windows NT ports.

By default, the list includes only standard ports controlled by the local print monitor, Localmon.dll. When you want to print over other communications channels (to a network-attached printer, for example), you must create a new port.

▷ **To create a new port**

1. Click the **Start** button, and then point to Settings.
2. Click **Printers.**
3. Click **Add New Port.**
4. Click to select the port monitor that controls the type of communications channel you want to use.
5. Insert the disk or type the network path to the port monitor files.
6. Click **OK** in each dialog box.

Print monitors often depend on other software components and appear in this list only when you have installed the components they require. For example, the Hewlett-Packard Network print monitor transmits print jobs by using the DLC network protocol. This monitor appears in the list only if you have installed the DLC protocol.

Each monitor displays its own user interface, which you use to create a new port and configure a printer to use it. To reconfigure the new port (if the monitor allows reconfiguration), click the **Configure Port** button on the Ports tab of the **Printer Properties** dialog box.

When you read about each print monitor in the following sections, remember that each is associated with a data communications channel, not with the print device at the other end of that channel. In most cases, the port monitor is unaware of the make or model of the print device it is communicating with, nor does it need this information. Also, although different port monitors might use the same network protocol, this does not make them interchangeable. For example, both the Digital Network Port print monitor and the LPR Port print monitor use the TCP/IP protocol, but they send data over that protocol in very different ways.

Local Print Monitor (Localmon.dll)

The local print monitor, Localmon.dll, sends print jobs to local devices. These include familiar ports like LPT1 and COM1. Use of the less familiar FILE and Other ports are described below.

The FILE port appears in the default port list on the Ports tab of the **Printer Properties** dialog box. When you send jobs to a printer that uses this port, the local print monitor prompts you for the name of a file in which the print job should be stored.

If you select Other from the list of ports in the Print To box on the **Printer Properties** dialog box, and select the Local Port option, the local print monitor prompts you to enter a port name. Some possibilities include:

- An explicit filename, such as C:\DIR\FILENAME. All jobs sent to this port are written to the named file. Each new job overwrites the last one.
- The name of a print share, such as \\SERVER\PRINTER. Jobs sent to this port are transferred over the network to the named share by the network redirector. This can be useful if you need to send jobs to a network print server, but you want the job to spool locally as well as on the print server.
- The NULL port. Use this port to test whether network clients are able to send jobs. Simply pause the printer set to use this port, send a job from a network client, look at the printer in Print Manager to confirm that it arrived, and resume the printer. Jobs sent to NULL are simply deleted from the system without wasting paper or delaying real print jobs.

Hewlett-Packard Network Port Print Monitor (Hpmon.dll)

The Hewlett-Packard Network print monitor, Hpmon.dll, sends print jobs to HP JetDirect adapters. This includes both the network adapters commonly installed in print devices (such as the LaserJet 4 Si) and the JetDirect device (which connects a parallel print device to the network).

Although many JetDirect devices can communicate over several different network protocols, including DLC, IPX, TCP/IP, and AppleTalk, Hpmon.dll is specific to DLC; you must load the DLC protocol to use this print monitor.

The DLC protocol is bridgeable, but not routable. This means that if a Windows NT print server is on one physical subnet and a JetDirect device is on another physical subnet, the server can send jobs to the JetDirect if the two subnets are joined by a bridge but cannot send jobs if the two subnets are joined by a router.

The DLC protocol can be bound to multiple network adapters, but the HP print monitor software can only manage printers over one network adapter, and it must be either adapter 0 or adapter 1. If your Windows NT computer has multiple network adapters, make sure all the HP JetDirect-equipped printers are on the same physical subnet.

When you install, reinstall, or upgrade Windows NT, it assigns a number to each network adapter in your computer. The adapter with the lowest I/O base address is assigned the number 0; the adapter with the next lowest I/O base address is assigned the number 1, and so on. If you later add another network adapter, it is assigned the next unused number, regardless of its I/O base address. Later, if you upgrade or reinstall Windows NT, each adapter is again assigned a number based on its I/O base address. If the third adapter has a lower I/O base address than either of the other two, then the numbering changes. In this case, HPMON looks for JetDirect cards through the wrong network adapter on the server.

▷ To correct adapter numbering after an upgrade

1. Click Start, point to Settings, and then click Control Panel.
2. Double-click the Network icon.
3. Click the Adapters tab.
4. Remove each adapter that has an incorrect number.
5. Add the adapters back in the order that corresponds with their I/O addresses (the lowest address getting the lowest number).
6. Click OK.

You can configure ports managed by Hpmon.dll as either *job-based* or *continuous connection*. The setting affects all ports at once:

- With job-based connection, a print server connects to the JetDirect adapter, sends a print job, and disconnects when the job is finished printing. This enables other print servers to connect and print.
- With continuous connection, a print server connects to the JetDirect adapter and maintains the connection until either the print server or the adapter is rebooted, preventing other servers from connecting. The main advantage of continuous connection is that all users are validated by the Windows NT security model, and every print job access can be audited.

  For more information about auditing printing, see the *Windows NT 4.0 Workstation Resource Guide.*

Note If you configure two Windows NT print servers to send jobs to the same JetDirect device, configure both servers for job-based connections. If you configure one of the print servers for continuous connection, it prevents the job-based server from connecting to the print device.

Line Printer Port Monitor (Lprmon.dll/Lpr.exe)

Windows NT supplies a command line utility (Lpr.exe) and the LPR Port print monitor (Lprmon.dll). Both act as clients sending print jobs to an LPD service running on another computer. As mentioned previously, Windows NT also supplies an LPD service, the TCP/IP Print service, which receives print jobs sent by LPR clients, including UNIX computers and other Windows NT computers using the TCP/IP protocol.

To send print jobs, the LPR clients needs the network address of the LPD print server and the name that the LPD service associates with its print device. LPR sends print jobs to LPD along with instruction on how to process the job and the name of the print device.

When you add an LPR port while connecting to the LPR print server, you'll be asked in a dialog box to supply information identifying the print server (host) and the name of the printer. You can supply either the IP address or the host name of the server. The server can be one of the following:

- UNIX computer
- Windows NT computer
- Windows for Workgroups computer with a 3rd-party LPD
- network adapter, such as an HP JetDirect or Emulex NetJet

For example, let's say you're connecting to a printer named LABLASER on a UNIX computer whose IP address is 11.22.33.44, and whose name (defined in the hosts file on your Windows NT computer) is UNIXBOX. You can enter either "UNIXBOX" or "11.22.33.44" (without the quotation marks) in the Name Or Address Of Host Providing LPD box. You would enter LABLASER in the Name Of Printer On That Machine box.

To obtain the IP address of a network adapter, check the adapter's documentation. Some adapters provide a character-based application that you can access either by network connection or by directly connecting to the adapter by means of a serial-to-serial port cable.

If the server is a UNIX computer, and you don't know a valid name for the printer, you can often find it by looking at the /etc/printcap file on the UNIX computer. Each entry corresponds to a print queue ("printer" in Windows NT terminology) on the UNIX computer. Fields in these entries are separated by ":" characters, and for readability an entry can be broken over several lines by ending a line with a "\" character and beginning the next line with a space or tab character. The first field of each entry lists valid names for the queue, separated by "|" characters.

Continuing the LABLASER example, we might find entries like the following in the printcap file on the computer named UNIXBOX:

```
lp|lablaser|The_Lab_Printer:\
:lp=/dev/ttya:br#9600:\
:lf=/usr/spool/lpd/lablaser-err:\
:sd=/usr/spool/lpd/lablaser:
```

The first line in this example defines a print queue with three valid names: lp, lablaser, and The_Lab_Printer. You can use any of these names in the second field of the **LPR Port** dialog box previously shown.

Note This example is provided for illustrative purposes only. The UNIX system documentation is your best source of detailed information on your system's printcap file. Also note that the term "queue" is the UNIX term. Its Windows NT equivalent is "printer."

When the LPR Port print monitor receives the LPD server's network address and the proper queue name, it can send print jobs and processing instructions.

The LPR Port print monitor sends the **l** command by default, whereas the command line Lpr.exe utility sends the **f** command by default. With the Lpr.exe utility, use the **-o** command if you want to override the default on a job-by-job basis. To change the default command for a particular printer controlled by the LPR Port print monitor, you must modify a Registry parameter. Use the Registry Editor (Regedt32.exe) to find the following key:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print\Monitors\
LPRPort\Ports\<portname>\<IP Address or Host name>:<printer name>
```

In this key, add a value named PrintSwitch with type REG_SZ, and enter the control command you want to use. For instance, enter the letter "f" (without the quotation marks) if you want to use the **f** command by default.

Some UNIX computers do not follow the control commands alone when deciding how to process a print job. For instance, if you send an ASCII text file with an **l** command (which tells the spooler not to alter the print job) to a PostScript printer, it does not print correctly. Consequently, many UNIX systems have added software that converts scans jobs that arrive with the **l** command. If the scanner finds PostScript commands, the job goes directly to the printer. If no PostScript is found, the added software adds the PostScript code so that the job prints correctly.

If you send PostScript jobs from a Windows NT computer using LPR, and the printer controlled by the UNIX server prints the PostScript code instead of interpreting it, the UNIX server might have a scanner that does not recognize the output from the Windows NT PostScript driver as valid PostScript code. If this happens, you might need to reconfigure Windows NT to use the **o** control command by default.

If the client sends the **f**, **o**, or **p** control command, the Windows NT TCP/IP Print service assigns the TEXT data type to the print job—which tells the spooler to edit the job to make sure it prints. If the client sends the **l** control command, the Windows NT TCP/IP Print service assigns the RAW data type to the print job—which tells the spooler the print job needs no editing to print correctly. For more detail about data types, see "Data Types" earlier in this chapter.

Control commands are documented in the LPR specification, Request For Comment (RFC)1179, sections 7.17 through 7.29. For detailed information about RFC 1179, refer to the Windows NT Knowledge Base article, "Text of RFC1179 Standard for Windows NT TCP/IP Printing," reference number Q124734.

LPD-compliant print devices must supply one or more names for each of their print queues, even though these devices often have only one possible destination for the print jobs they receive. Each print-device manufacturer chooses naming conventions independently. Some, like the HP JetDirect adapter, accept any string as a legal print queue name. In comparison, the Emulex NetJet adapter accepts only two strings, and these are case sensitive, and default to TEXT and PASSTHRU. These strings are configurable. To find the names supported by any specific adapter, check its documentation, or contact the vendor's technical support group.

Remember that the local print provider spools print jobs, and that it despools them to the print monitors. In Windows NT, after LPRMON receives a job from the local print provider, it spools it a second time as a temporary file in the SYSTEM32 subdirectory. LPR must include an accurate byte count in the control file, and it cannot get that byte count from the local print provider. Instead, it respools the data to a temp file, finds the size of that temp file, and sends that size in the control file to the LPD server.

The most common problem people encounter when printing from UNIX systems to a Windows NT print server is that their print jobs are processed as TEXT data type instead of RAW data type, as they would be on a UNIX system. This happens because the UNIX systems almost always send the **f** control command, expecting that the control command isn't too important, because the TCP/IP (LPD) server parses the job to identify what data type to use and how to alter the job. However, Windows NT relies on the control command to determine the data type. As a result, the LPD service on Windows NT assigns the TEXT data type to most jobs.

To correct this problem, reconfigure the LPR client to send the l control command so the LPD will assign the RAW data type.

Macintosh Port Monitor (Sfmmon.dll)

The Macintosh port monitor, Sfmmon.dll, transmits jobs over a network using the AppleTalk protocol to network-attached print devices, such as the Apple LaserWriter family. It also lets you send jobs to AppleTalk spoolers, regardless of the print device that the spooler is attached to.

This monitor is available on both Windows NT Workstation and Windows NT Server computers and enables any Windows NT-based computer to send local print jobs to AppleTalk print devices. However, only Windows NT Server has a Macintosh print server component, so only a Windows NT Server computer can receive print jobs from Macintosh clients.

Some print devices process non-PostScript print jobs incorrectly if they receive those jobs over AppleTalk. Also, some print devices process PostScript jobs incorrectly if those jobs contain binary data and arrive over any protocol other than AppleTalk. These problems result from restrictions in those print devices; they do not indicate that the Windows NT Server print server is transmitting the jobs incorrectly.

Digital Network Port Monitor (Decpsmon.dll)

The Digital Network Port print monitor, Decpsmon.dll, sends print jobs to Digital Equipment Corporation's Digital PrintServer print devices and to other Digital Equipment Corporation print devices (such as the DEClaser 5100 and the DECcolorwriter 1000).

Windows NT supplies the TCP/IP network protocol, but does not supply the DECnet™ protocol. If you want to use DECnet, you must contact Digital Equipment Corporation to obtain it.

LexMark Mark Vision Print Monitor (Lexmon.dll)

LexMark's Mark Vision print monitor (Lexmon.dll) sends print jobs to LexMark MarkNet print devices (such as the MarkNetXL) and to network adapters (such as the MarkNetXLe). This print monitor uses any of the following protocols:

- DLC
- TCP/IP
- IPX

PJL Monitor (Pjlmon.dll)

The PJL Language Monitor, which Windows NT 4.0 supplies, "speaks" printer job language (PJL). Any bi-directional print device that uses PJL can use PJL Language Monitor. For example, PJL is the language that implements all the bi-directional communication between an HP LaserJet 5Si (a bi-directional print device) and its print server.

Bi-directional Printing

Bi-directional communication allows applications to query the print device directly to determine its capabilities. It also provides the benefit of configuring printer driver settings on the server without user intervention. The printer driver can automatically determine how much memory the printer has, what printer fonts are available, and so on.

Bi-directional print devices can send unsolicited messages to Windows NT and to applications. For example, the print device might send an "out of paper" or "printer offline" message. Bi-directional communication enables much more detailed status reporting on a wider variety of information, such as low toner, paper jams, maintenance needs, and so on.

To take advantage of bi-directional printing, you must have the following:

- A bi-directional print device.
- An IEEE 1284-compliant cable (a cable that has "1284" printed on it).
- A correctly configured port. (Some parallel ports are set to IBM AT-compatible mode by default; in this case, you need to set the port to IBM PS/2 mode.)

Printer Properties

Printer properties include everything about a printer—from what port you use to what security features you want to implement. For detailed information, see "Setting Up Print Servers" in *Microsoft Windows NT Server Concepts and Planning*. For procedural information, see Help.

▷ **To set or change printer properties**

1. Click Start, and then point to Settings.
2. Click Printers.
3. In the Printers window, click the printer to select it.
4. Click **Properties** on the **File** menu.
5. On the Printer Properties sheet, click the tab you want.

   The tabs are described in the following table.
6. Click **OK**.

Tab	Description
General tab	Displays general information about the printer, such as its physical location, a descriptive comment, and the printer driver installed. You can also add or customize a separator page, select a vendor-supplied print processor, and print a test page.
Ports tab	Displays the computer's physical ports. You can also add, configure, and delete virtual ports.
Job Scheduling tab	Displays the times the printer is available, its priority level, and spooling options.
Sharing tab	Displays the share status and share name.
Security tab	**Permissions** button lets you set the level of access for different types of users. **Auditing** button lets you set up printing users and events to be tracked. **Ownership** button displays printer owner.
Device Settings tab	Displays everything from paper tray sizes to font substitutions table for that printer. Use it to download soft fonts, install font cartridges, increase printer virtual memory, etc.

Separator Page Files

Separator pages—also called burst pages, header pages, or banner pages—print before each print job. They typically identify the computer that created the job, the time it was created, and so on. The local print provider contains an interpreter that reads commands from a separator page file to produce separator pages.

By default, separator page files are stored in the *%systemroot%*\SYSTEM32 directory. To use a separator page file, click the **Separator Page** button on the General tab of the printer's Printer Properties sheet. Either type the name of the Separator Page file, or browse through the directories and select a file. Windows NT provides three separator page files; you can use them as is, or modify them to create your own custom separator page files.

The following table lists the separator page files included with Windows NT.

Filename	Purpose	Compatible with
Sysprint.sep	Prints a page before each document	PostScript
Pcl.sep	Switches dual-language HP printer to PCL printing	PCL
Pscript.sep	Switches dual-language HP printer to PostScript printing	PostScript

To create a custom separator page file, copy and rename one of the supplied files, and then edit it. The following table provides the escape codes you can use. The separator file interpreter replaces these escape codes with appropriate data, which is sent to the printer.

Escape code	Function
\	The first line of the separator file must contain only this character. The separator file interpreter considers this the separator file command delimiter. This table assumes that character is a \ character.
\N	Prints the user name of the person who submitted the job.
\I	Prints the job number.
\D	Prints the date the job was printed. The representation of the time is the same as the Time Format in the International section on the Control Panel.
\L*xxx*	Prints all the characters (*xxx*) following it until another escape code is encountered.
\F*pathname*	Prints the contents of the file specified by *pathname*, starting on an empty line. The contents of this file are copied directly to the printer without processing.
\H*nn*	Sets a printer-specific control sequence, where *nn* is a hexadecimal ASCII code sent directly to the printer. To determine the specific numbers, see your printer manual.
\W*nn*	Sets the width of the separator page. The default width is 80 characters; the maximum width is 256. Any printable characters beyond this width are truncated.
\B\S	Prints text in single-width block characters until \U is encountered.
\E	Ejects a page from the printer. Use this code to start a new separator page or to end the separator page file. If you get an extra blank separator page when you print, remove this code from your separator page file.
\n	Skips *n* number of lines (from 0 through 9). Skipping 0 lines simply moves priting to the next line.
\B\M	Prints text in double-width block characters until \U is encountered.
\U	Turns off block character printing.

Print Security

To limit user access to your network printers and spool file directory, use Windows NT security. You can provide additional security by managing print jobs from Macintosh clients or by forwarding print jobs to other print servers.

Printer Security

To specify security attributes on a printer-by-printer basis, use the Security tab in the **Printer Properties** dialog box. The most efficient way to establish security attributes is to assign permission levels to different user groups. For example, you could give all nonadministrative users in a department the Print level of permission and all managers Full Control permission.

The following table summarizes printer security permissions.

Type of permission	Level of access
Full Control	Enables complete access and administrative control.
Manage Documents	Enables a person to change the status of any print job submitted by any user. Does not permit control of the printer status.
Print	Allows user to send print jobs to the printer and to control pause, resume, or delete for his or her own jobs.
No Access	Explicitly denies access to a specific printer.

The installation-default printer permissions are different for Windows NT Server and Windows NT Workstation computers. By default, the following print permissions are assigned on a Windows NT Server computer:

- Full Control permission for the Administrator, Server Operator, and Print Operator groups
- Manage Documents permission for the Creator Owner group
- Print permission for all users

▷ **To change security permissions**

1. Click Start, and then point to Settings.
2. Click Printers.
3. Click to select the printer whose settings you want to change.
4. Click **Properties** on the **File** menu.
5. Click the Security tab.
6. Click Permissions.
7. Click to select the group whose permission you want to change, and then select the permission level from the Type Of Access dropdown list.

   To add a group, click the **Add** button. To remove a group, select it, and then click the **Remove** button.
8. Click **OK** in each dialog box.

Security for Macintosh Clients

Although native Macintosh networking provides support for file security, it does not provide support for print device security. The AppleTalk protocol contains no mechanism that supports client user name or password. Macintosh clients, therefore, cannot identify themselves on the network, and the Windows NT print server cannot impose user-level security on Macintosh clients. If a Macintosh client is physically able to send a job to a print device or print server, that client implicitly has permission to do so.

You can, however, enforce one set of printer permissions on all Macintosh users as a group. The Macintosh client must start the MacPrint service by logging on using a user account. By default, it logs on as the System account. The System account has Print permission on all local print devices, so any Macintosh client can send a job to any of the Windows NT computer's local printers. To limit the permissions for Macintosh clients, create a new user account, giving it the printer permissions you want the group to have. Then set the Macintosh client MacPrint service to log on using this account.

Note The System account on one computer does not have permission to access resources on other computers. Macintosh clients that start the MacPrint service by logging on as System user cannot send jobs to printers that forward jobs to other print servers. The solution is to configure the Macintosh client MacPrint service to log on as another user, one which has permission to print on all the print servers to which print jobs are forwarded.

Spool File Security

To implement spool file security, change the default spool directory to an NTFS partition and directory where write permission is limited on a user-by-user basis.

When a print job prints locally, the local print provider spools the job to disk during processing. By default, the Everyone group has Change permission in the default spool directory. This allows all user print jobs write access to the default spooler directory.

A print job that cannot be spooled to disk during processing does not print. If the spool directory location is changed, all users who need to print must have Change permission for the new spool directory.

As described in the "Local Print Provider" section, the spool file (SPL_) and the shadow file (SHD) are written to the default spooler directory:
%WINNT\SYSTEM32\SPOOL\Printers

For instructions on changing the default spool file directory, see "Local Print Provider" ealier in this chapter.

▷ **To override the default location for one specific printer**

1. Start the Registry Editor (Regedt32.exe).
2. Find the following key:

   ```
   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
   Print\Printers
   ```

3. Find the key for the printer:

   ```
   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
   Print\<Printername>
   ```

4. Add a new SpoolDirectory setting, and as its value provide the path to the spool directory that this printer should use.

Note You must create a directory for the new spool file location. If you attempt to spool directly to the root (C:\ or D:\, for example), the spool file will revert to the default spool directory.

The change in the Registry takes effect after you stop and restart the Spooler service.

Registry Security

Most printing-related Registry settings reside in the following subkey:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print
```

An administrator can use the Registry Editor to assign read-only access to these subkeys. Users with read-only access cannot add or configure printers, because the read-only access does not allow changes of these subkeys.

Windows-based applications also use the following subkey in the Registry to find information about available printers.

```
HKEY_CURRENT_USER\Software\Microsoft\WindowsNT\
CurrentVersion\PrinterPorts
```

Users with read-only access to this subkey cannot add new printers that are recognized by Windows-based applications.

Forwarding Jobs

To forward a print job to a different Windows NT print server, a Windows NT print server uses a null session. By default, the null session is disabled, preventing job forwarding. To enable job forwarding and null-session support, manually edit the following Registry subkey:

```
\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\
LanmanServer\Parameters
```

In this subkey is a value named NullSessionShares. Add a new line containing the share name for the printer. This change does not take effect until you stop and restart the Spooler service.

Troubleshooting Printing Problems

Troubleshooting Windows NT printing problems can be a challenge because of the number of variables involved in printing and the number of different clients and print devices that Windows NT supports.

Windows NT has a modular printing architecture, with a module for each major task (such as receiving jobs from network clients). By adding new modules you can easily add functionality (such as support for a new type of network client). This modularity gives Windows NT a great deal of flexibility, making it able to support a wide variety of client operating systems, applications, data objects, network configurations, spooling options, and print devices. That flexibility comes at a cost, however, because each additional configuration adds its own possible points of failure.

Successful troubleshooting depends on your ability to quickly rule in or rule out general categories of points of failure. The modularity makes this fairly easy: Network printing, for example, consists of seven processes that always occur in the same order. By testing one of the processes, you can determine whether the problem is occurring in that process, before it, or after it.

The following are the seven basic processes involved in a network printing job:

1. An administrator creates a print share on the print server.
2. A client system connects to that share.
3. The client system creates a print job.
4. The client system sends the print job to the print share on the print server.
5. The print server receives, spools, and sometimes modifies the print job.
6. The print server sends the job to the print device.
7. The print device interprets the job and produces hardcopy output.

The basic strategy for troubleshooting printing problems is to use problem symptoms to identify the process, or processes, that are creating the problems. If you still cannot correct the problem after you've located it, your Product Support engineer will be much better able to help you correct it if you have first isolated the process where the problem exists.

▷ **To troubleshoot printing problems**

1. Identify which of the seven processes is failing.
   - Analyze symptoms to identify the most likely process.
   - Reconfigure that process.
   - Retest with the new configuration.
   - If the problem changes with the new configuration, you probably have the right process. Otherwise, you probably picked the wrong process.
2. Look for documented explanations or solutions from the following sources.
   - Product hardcopy documentation
   - Product online Help documentation.
   - Microsoft Product Support Knowledge Base.
3. If possible, reconfigure the process to avoid the problem in the short term.
4. Implement long term solutions.

To help you implement this strategy, this section presents the seven basic printing processes and gives the following information about each process:

- A basic description of what the process involves
- A list of variables that can affect the process
- Symptoms that would suggest a problem in this process
- Tests to prove or disprove the suspicion

Printer Definition and Configuration

A print server administrator can use the Add Printer wizard to:

- Create a local printer.
- Connect to a remote printer.

Both procedures install and configure a printer driver. Creating a printer gives more control over driver configuration than connecting to a printer, and the administrator can create new printer ports by entering values in the **Print To** field on the Printer Properties sheet. The administrator can also define form-to-tray mapping, security, and spooler options.

The problem is likely in this process if...

- You cannot launch the Add Printer wizard, or you cannot browse printers in the Add Printer wizard.
- You cannot create a new local printer or a new local printer port.
- The port name and port do not show up or are intermittent.
- The problem is specific to one printer or to a subset of the printers.
- You can create another printer (with the same driver, port, and configuration options, but a different name) and the problem does not occur with that new printer.
- You cannot install a particular printer driver, or the driver for that printer doesn't exist.
- You cannot share a printer.
- You cannot establish security on printers.
- You have problems with forms, separators, pages, fonts, halftones, or printer options.
- You can delete the printer, stop and restart the spooler, recreate the printer with the same name and configuration, and the problem no longer occurs.

Client Computer Connects to a Shared Printer

A printer is established on a print client computer.

The problem is likely in this process if...

- You have problems with forms, separators, pages, fonts, halftones or printer options.
- You can connect to another shared printer, on the same print server, and the problem goes away.
- You cannot connect to a remote printer.
- You cannot browse printers in the Add Printer wizard.

Client Application Creates a Print Job

A user on a client system runs an application that composes text and/or graphics to create a print job. The application may interact with a printer driver to create output in a printer language such as PCL, PostScript, or HP-GL/2.

The problem is likely in this process if...

- The error in one particular print job is not reproducible.
- You create a simple test document and print it to a file. Transfer that file to a different client system, and print it to a different printer on a different print server, and there it continues to fail.
- What you see on the screen is not the output that you get.
- The problem is limited to one particular user or the users in a specific group; one particular client operating system; a particular vendor's printer driver (or a particular version of the printer driver); a particular font or fonts from a particular vendor; certain extended ASCII characters; a particular graphic object, a particular graphic file format (such as EPS, BMP, or TIF); jobs created by a particular application or version of that application; color or shading; incorrect resolution (dots per inch);, or the print job is missing line feeds or form feeds.
- A different driver works fine.

If the problem persists when you send a simple textprint job from several clients' systems, the problem is likely caused by another printing process.

Client Sends Job to Spooler

The user on the client system sends the job over the network to the print server. The client system's application software or operating system sends the job to the client's transport protocol, to the network adapter, over the network hardware, to the transport software on the print server, and finally to the appropriate print server service on the print server. The print server service (Windows NT Server, Services for Macintosh, or TCP/IP Print) assigns a data type to the print job and submits the print job to the spooler, or leaves it blank.

The problem is likely in this process if the problem is limited to...

- One particular client redirector (one vendor's software, or one version of that software).
- One particular network transport—for example, TCP/IP, NWLink, NetBEUI, AppleTalk.
- One particular make or model of network adapter, or one firmware level.
- One particular intermediate system for example, one specific router, bridge, hub, or gateway.
- One particular kind of intermediate system, for example, all jobs from clients on the other side of any router, bridge, or gateway.
- One particular kind of client, for example MS Network, LPR, or Macintosh clients.

The problem is also likely in this process if...

- The print job is not started until the application is exited.
- Pages come out incomplete.
- The print server has run out of disk space.

Suspect a problem elsewhere if...

- The print job prints fine on another printer of the same make, model, or version.
- You send a print job to the spooler in a different way, and the problem does not go away.

Print Server Spooler Processes Print Job

The spooler receives the job from the print server service. If the printer was established by selecting the **Network Printer** option in the Add printer wizard, the remote print provider sends the job to the print server. Otherwise, the job goes to the local print provider.

Unless otherwise configured, the local print provider spools the job to disk and checks the data type assigned by the print server service. The data type (for example, RAW or PSCRIPT1) determines which print processor receives the job. The data type also effects whether the print processor alters the job or not, and if so, how it alters the job. If the spooler is configured to append a separator file, it does so before sending the job to the print monitor for delivery to the print device.

The problem is likely in this process if...

- Page size, font, or character set is wrong because of incorrect data type (UNIX).
- Disk space is limited.
- You get extra form feeds or no form feed.
- Changing the default data type affects the problem with MS-Network.
- The job gets stuck in the printer.
- The problem is specific to one type of client.
- Macintosh jobs print in the wrong resolution, have font problems, or print black and white when they should print in color.
- The problem occurs only when the printer was established using the Add Printer wizard.

Print Server Spooler Sends Job to Print Device

The print monitor receives the job and interacts with local hardware drivers and transport drivers to send the job to its destination. The components in this process are:

- Print monitors, LOCALMON, redirector, LPRMON, SFMMON, DECMON, or HPMON.
- Transport protocols (either TCP/IP, NWLink, NetBEUI, or AppleTalk).
- Network hardware (routers, bridges).

The problem is likely in this process if it is limited to...

- All print devices accessed by a particular print monitor.
- All print devices on one segment of the LAN.
- One make/model/version of a particular print device.
- One parallel/serial cable.

The problem is also likely in this process if...

- Sending the same job to the same device by means of a different communication channel by using a different protocol, serial, or parallel port produces a successful print.
- Trying different parallel or serial cable produces a successful print. If your print device is bi-directional, make sure that the cable has "1284" printed on it. The cable should be six to 15 feet (maximum).
- If your print device is bi-directional, make sure that the parallel port is set correctly. Some are set to AT-compatible mode by default, which is not bi-directional. Set the port to PS/2 mode in the BIOS to get bi-directional functionality.

Print Device Interprets Job

The final processing at this point is completed by the print device. The print device receives the print job from a hardware port. The print device interpreter interprets the job and produces hardcopy output.

The problem is likely in this process if...

- The problem is specific to a specific print device, or a print device that is a variation of the same model, make, or revision.
- Sending same print job to another print device of same make, model, and revision produces a successful print.

Questions and Answers

The following section presents specific questions and answers about Server Printing.

Does the Windows NT print server support UNIX clients running LPSSCHED?

No, it does not support UNIX computers running LPSSCHED. UNIX client computers must have an LPD program installed when used with a Windows NT print server.

How can platform-specific printer drivers be installed for a print client on a hardware platform different from the Windows NT print server?

Microsoft places all new or updated printer drivers onto its electronic services for public download.

Also, a platform-specific printer driver is likely available on another computer in the network.

▷ **To install platform-specific printer drivers from another computer in the network**

1. Log on the client computer using an user account that has Full Control permission.
2. Double-click Network Neighborhood.
3. Double-click the print server.
4. Click to select a printer.
5. Click **Properties** on the **File** menu.
6. Type the location of the printer driver in the **Printer Properties** dialog box. The printer driver is installed on the print server where it is available to any client computer that has the same hardware platform and is connected to the Windows NT print server.
7. Click **OK**.

What kind of problems occur when a print job has been assigned an incorrect data type?

Typical problems that occur when there is some error in the data type assignment and consequent job alteration of a print job include the following:

- LPR client print jobs include PCL or Postscript code, include incorrectly printed extended characters, or print in the print device's default font.
- The final page of a Microsoft network-based client print job does not print.
- Extra page prints after a Microsoft network client print job.
- Microsoft Network client print jobs include PCL or Postscript code, include incorrectly printed extended characters, or print in the print device's default font.
- Postscript print jobs sent from Macintosh clients do not print in color, print at a lower resolution, or fail.

How many printers can be supported by a Windows NT print server?

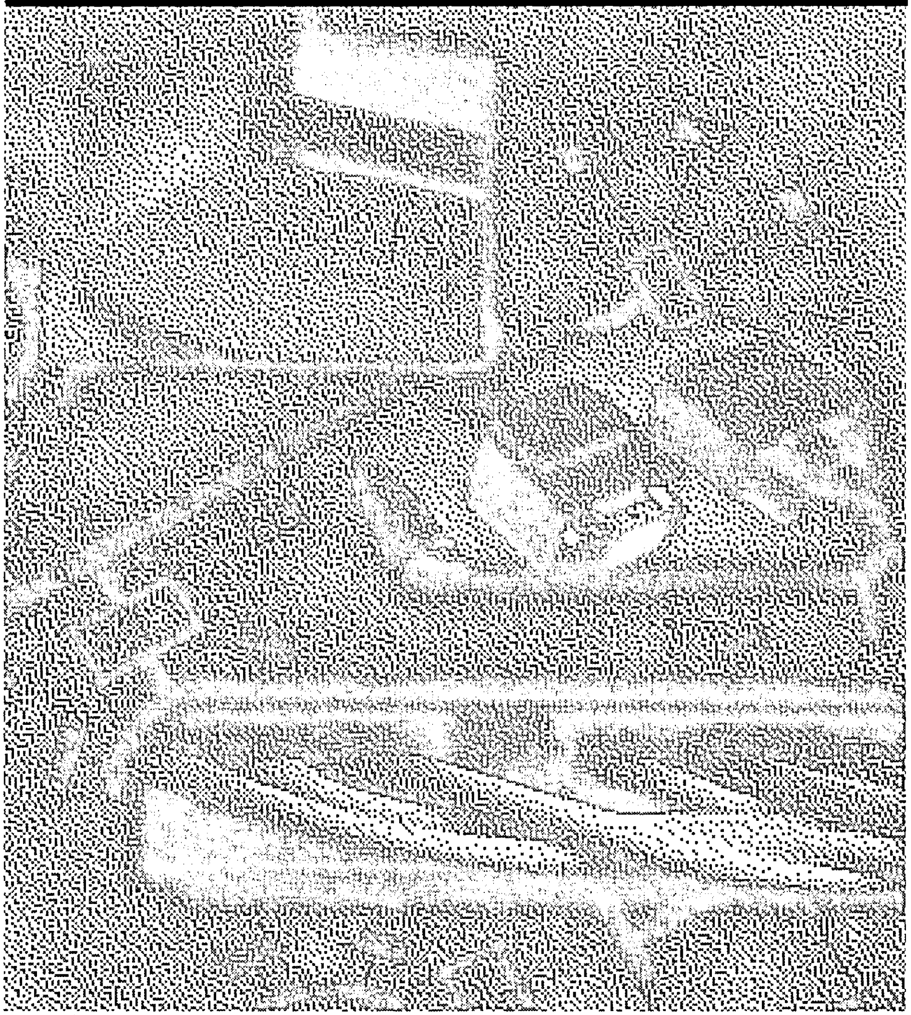
The limitation on the number of printers that can be attached to Windows NT print server is dependent on whether the print server is a Windows NT Workstation computer or a Windows NT Server computer. Windows NT Workstation and Windows NT Server have been optimized for different roles in the network:

- Windows NT Workstation is limited to 10 connections from other computers and should be used as a print server in small-network situations.
- Windows NT Server has been optimized as a print, file, and application server.
- Windows NT versions 3.5 and 3.51 are capable of supporting more printers than Windows NT version 3.1.

The limitation on total printers attached to a Windows NT Server print server should be determined by daily requirements for print throughput. Print throughput is determined by the processor capabilities. For example, a print server with approximately 64 meg RAM and a 486/66 or higher processor with a high throughput network card can support 25 to 30 DLC printers or 40 TCP/IP printers.

PART II

Reliability and Recoverability



CHAPTER 3

Disk Management Basics

This chapter is about managing and understanding disks. Its purpose is to provide you with a basic understanding of disk hardware and how Windows NT accesses the disks. It also describes the Windows NT startup process, so you know what is happening when you power on your computer. We want you to be able to effectively troubleshoot disk and startup problems.

Why You Should Know About Disks

Disk problems are frequently the cause of startup failures. Some of these problems are the result of hardware errors. But frequently the problem is that Windows NT is unable to find the information about the disk or its contents, usually because of control information corruption. If you know what the control information on your disk should be, you can tell if it has been corrupted.

There are several reasons that Windows NT cannot give your computer complete protection:

- The security provided by Windows NT is effective only when the operating system is running. If you are running another operating system, or if an unauthorized person can walk up to the computer and restart it, Windows NT cannot provide any protection.
- External media devices, such as floppy disks and CD-ROMs, provide the physical means for anyone to bypass Windows NT and gain access to your files. You can also introduce viruses from floppy disks or from files that you copy or download from other computers.
- The transaction logging that the Windows NT file system (NTFS) does for recoverability purposes protects only NTFS metadata.
- A power failure can cause corruption. So can turning off your computer without going through shutdown.

This following list highlights simple procedures that you can implement to reduce the potential for problems and make it easier for you to recover when problems occur.

- First, implement physical security. Put your computers in a secure room. Lock them. Use a password on your screen saver. Run virus checks on your computer and on floppy disks before you use them. Or disable the floppy disk, which is supported in some system BIOSs. If you cannot disable the floppy disk in the BIOS, you can physically disconnect it.
- Develop a preventive maintenance plan and schedule regular maintenance. You might not know about file system problems until you restart your computer, when the Chkdsk program tells you that there are errors or that a volume is Unknown.
- Back up data that are critical to the operation of the computer when you make changes to them. These data include the Registry, Master Boot Record, and Partition Boot Sectors. You also need a way to start another installation of Windows NT Server if the one you normally use doesn't work, either because hardware failure or corruption makes the primary installation unusable.
- Know the configuration of the computer. If you have more than one disk with more than one partition, you need to know which data are on which partitions on which disks. For example, which partition is your system partition or your boot partition? If you have implemented volume sets, stripe sets, mirror sets, or stripe sets with parity, which disks are members of which sets?
- Implement a backup procedure for user data. Users will delete files or make changes to them and then want the old version back. Files can get corrupted when applications crash. A power failure can adversely affect files and disks in many ways. You need to be able to restore individual files, as well as a complete disk's worth of data, or even all of the files on the computer.

This chapter, and the other chapters in the "Reliability and Recoverability" part of this book, provide much more detailed information about the preceding topics. There are more details about disk organization, file systems, and starting your computer in the following chapters of the *Windows NT Workstation Resource Guide*:

- Chapter 17, "Disk and File System Basics."
- Chapter 18, "Choosing a File System."
- Chapter 19, "What Happens When You Start Your Computer."

About Disks and Disk Organization

This section describes the physical structure of hard disks and floppy disks, and presents an overview of ways to logically organize areas on your disks.

Both hard disks and floppy disks work similar to an old-fashioned record player, with the record turning on the turntable and a needle playing the music. They are also similar to VCRs and stereo cassettes in that disks, cassette players, and VCRs all use a read/write head to access the information that is stored on the magnetic media.

Hard or fixed disks store information on a revolving platter of metal or glass coated with a magnetic material. The disk typically consists of several physical platters on a common spindle. This platter is similar to the record on a record player, except information is usually stored on both sides of a disk platter.

As the disk rotates, a transducer element called the head reads or writes binary data on the magnetic media. There are many methods for encoding data on the disk, such as modified frequency modulation (MFM) and run length limited (RLL). The disk controller logic determines the method used and the density of data on the disk.

Note There are some state-of-the-art disks and disk controllers that have different hardware and logic than are described in this chapter.

Hardware Terminology

Each disk consists of platters, rings on each side of each platter called tracks, and sections within each track called sectors. A sector is the smallest physical storage unit on a disk, almost always 512 bytes in size.

Figure 3.1 illustrates a hard disk with two platters. The remainder of this section describes the terms used on the figure.

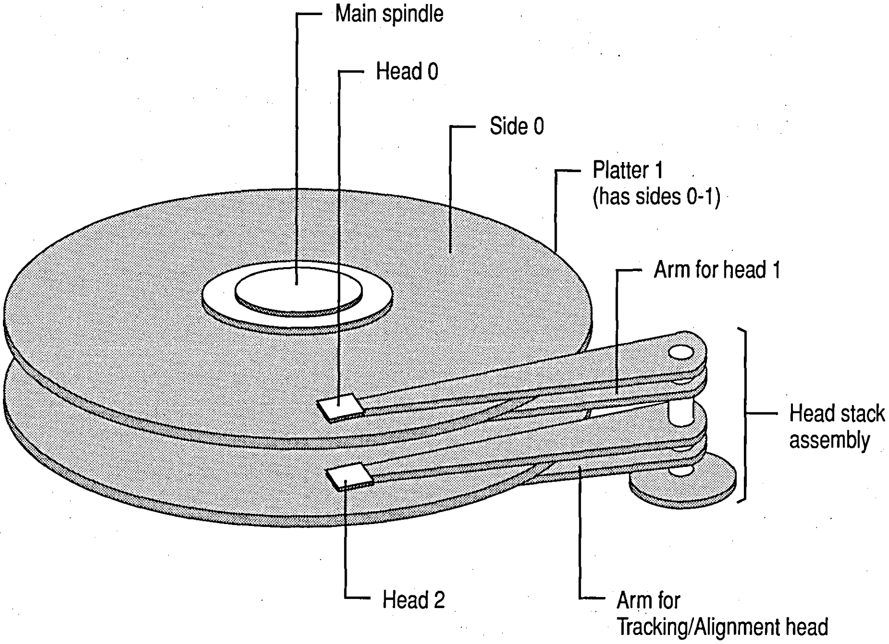


Figure 3.1. Illustration of a hard disk

The cylinder/head/sector notation scheme described in this section is slowly being eliminated. All new disks use some kind of translation factor to make their actual hardware layout appear as something else, mostly to work with MS-DOS and Windows 95. See the section titled "Translation," presented later in this chapter, for more information.

Tracks and Cylinders

On hard disks, the data are stored on the disk in thin, concentric bands called tracks. There can be more than a thousand tracks on a 3½ inch hard disk. Tracks are a logical rather than physical structure, and are established when the disk is low-level formatted. Track numbers start at 0, and track 0 is the outermost track of the disk. The highest numbered track is next to the spindle. If the disk geometry is being translated, the highest numbered track would typically be 1023. Figure 3.2 shows track 0, a track in the middle of the disk, and track 1023.

A cylinder consists of the set of tracks that are at the same head position on the disk. In Figure 3.2, cylinder 0 is the four tracks at the outermost edge of the sides of the platters. If the disk has 1024 cylinders (which would be numbered 0-1023), cylinder 1023 consists of all of the tracks at the innermost edge of each side.

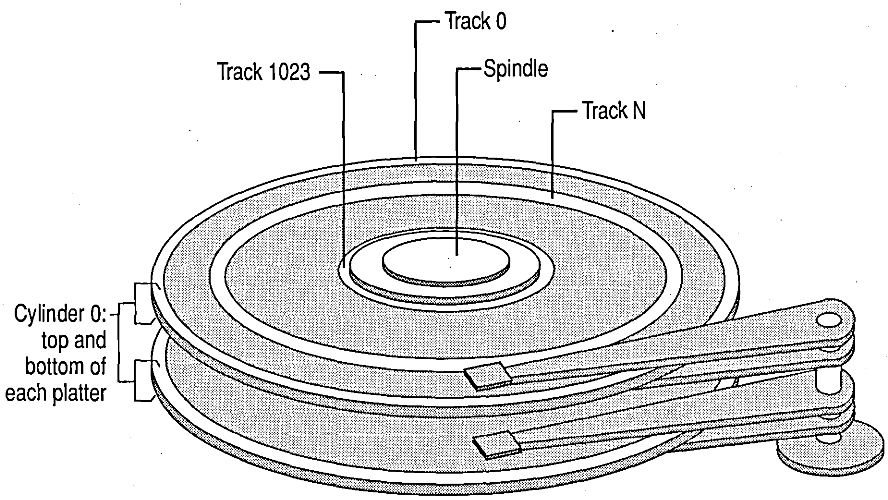


Figure 3.2. Tracks

Heads and Sides

A head is a small transducer that can be positioned over a disk track by using a solenoid or servomotor. The head can change the properties of the magnetic media as it rotates underneath.

There is one head for each side of each platter. The heads are normally attached to a common head-movement mechanism, so that the heads all move in unison. The heads are always positioned at the same logical track on each side of each platter.

Because disk tracks are concentric bands, the heads must move to the track that contains the data to be accessed. The moving of the heads from the current track to the track that contains the next data is called seeking. The heads are moved over the surface of the disk in small increments called steps. Each step corresponds to one track.

Sectors and Clusters

Each track is divided into sections called sectors, and each track on the disk has the same number of sectors. A sector is the smallest physical storage unit on the disk. The data size of a sector is always a power of two, and is almost always 512 bytes.

The Windows NT file systems allocate storage in clusters, where a cluster is one or more contiguous sectors. Windows NT bases the cluster size on the size of the volume.

As a file is written to the disk, the file system allocates the appropriate number of clusters to store the file's data. For example, if each cluster is 512 bytes and the file is 800 bytes, two clusters are allocated for the file. Later, if you update the file to, for example, twice its size (1600 bytes), another two clusters are allocated.

If contiguous clusters (clusters that are next to each other on the disk) are not available, the data are written elsewhere on the disk, and the file is considered to be fragmented. Fragmentation is a problem when the file system must search several different locations to find all the pieces of the file you want to read. The search causes a delay before the file is retrieved. A larger cluster size reduces the potential for fragmentation, but increases the likelihood that clusters will have unused space.

Using clusters larger than one sector reduces fragmentation, and reduces the amount of disk space needed to store the information about the used and unused areas on the disk. Because the FAT file system can use only 16 bits for the cluster number, using clusters enables FAT volumes to be larger than 65,535 sectors. There is more information about the FAT file system and FAT volumes presented later in this chapter.

Figure 3.3 shows a sector and a grouping of four contiguous sectors into a cluster.

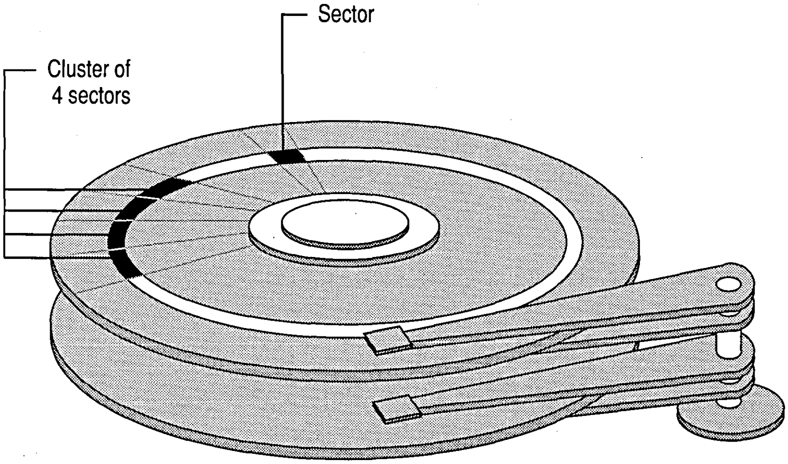


Figure 3.3. Sectors and clusters

Translation

Translation is the conversion of physical disk geometry (the number of cylinders on the disk, number of heads per cylinder, and number of sectors per track) to a logical configuration that is compatible with the operating system.

Many state-of-the-art disks now have a feature called Translation Mode. The disk queries the system BIOS (x86-based computers) or firmware (RISC-based computers) to determine the logical parameters the computer needs. If the number of cylinders, heads, and sectors per track the computer needs is within the range supported by the disk, the disk automatically configures itself to match those parameters. If the system BIOS or firmware does not support translation, the disk configures itself to its largest possible size, which is typically around 500 MB.

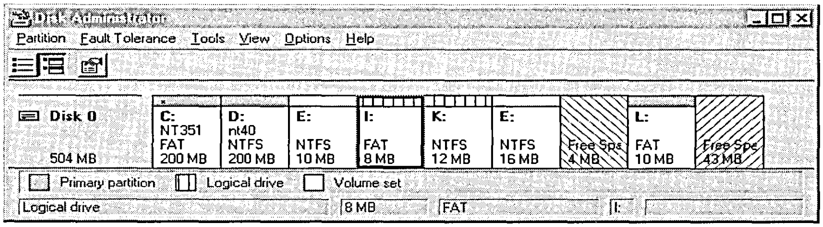
For example, all SCSI disks use some kind of translation, and many SCSI disks support hot swapping of sectors to fix·bad sectors. This means that a SCSI disk maintains a set of unused sectors at the end of a disk. When a used sector goes bad, it is remapped to an unused sector and the data are written to the new sector. Windows NT uses absolute sector notation (also called logical sector notation, or LSN) for accessing disks. The translation of the sector number to the physical sector on the disk is handled by low-level disk device drivers and the hardware.

The translation is especially important when there are more than 1024 physical cylinders on the disk. The section titled "Using Hard Disks With More Than 1024 Cylinders (x86-based Computers)," presented later in this chapter, contains more details. There is also more information in Chapter 5, "Preparing for and Performing Recovery."

Logical Organization: Partitions and Volumes

The Master Boot Record on each hard disk contains an area called the Partition Table that the computer uses to determine how to access the disk. It has room for four entries, called partitions, that you create to make it easier to organize information. A partition must be completely contained on one physical disk.

Use Disk Administrator to create and format partitions. The next figure is a Disk Administrator screen shot for a computer with a single disk.



Disk 0 504 MB	C: NT351 FAT 200 MB	D: nt40 NTFS 200 MB	E: NTFS 10 MB	I: FAT 8 MB	K: NTFS 12 MB	E: NTFS 16 MB	Free Spa 4 MB	L: FAT 10 MB	Free Spa 43 MB

Logical drive 8 MB FAT I:

Figure 3.4. Disk Administrator screen shot of primary partitions and an extended partition

Disk Administrator shows you the layout of each of your disks, including:

- The size of each disk.
- The size, label, and file system used for each partition on the disk.
- The size and location of free space.

The remainder of this section describes partitions, and explains what you see in the Figure 3.4.

There are two types of partitions in the Master Boot Record, primary and extended. A primary partition is one into which you can install the files needed to load an operating system. A primary partition is formatted for a particular file system and is assigned a drive letter. Having multiple primary partitions enables you to install and start operating systems that do not use the same file system, such as Windows NT and UNIX. There are three primary partitions in Figure 3.4, C, D, and L.

You can have one extended partition on a hard disk. An extended partition is effectively a logical disk. Unlike a primary partition, you do not format the extended partition, nor does it get assigned a drive letter. Instead, you can create one or more logical drives within the extended partition, and each logical drive is assigned a drive letter. You format each logical drive for a particular file system.

An extended partition is a method for avoiding the four-partition limit and configuring a hard disk into more than four logical areas. An extended partition entry looks much like a primary partition entry in the Master Boot Record, except that it points to space on the disk that can contain one or more logical drives. The important information about an extended partition is how much of the disk it takes up and where it starts.

Note In this chapter, and other chapters in the "Reliability and Recoverability" part of this book, the term logical drive means a contiguous area in an extended partition, which is assigned a drive letter.

There is an extended partition in Figure 3.4, but it is not as easy to identify as the primary partitions. The extended partition consists of all the disk area between primary partitions D and L. If there are logical drives on a disk, then you have an extended partition on that disk. If you have free space for which the diagonal lines look like the ones for the 4 MB of free space to the left of primary partition L, then you have an extended partition on the disk.

Figure 3.4 shows an important point. The unpartitioned space at the end of the disk (to the right of primary partition L) cannot be used. You can have only three primary partitions and one extended partition on a hard disk, or four primary partitions. This disk already has the maximum for each of these types.

Note The distinction between the two areas of free space is that the one to the left of L is free space within an extended partition, while the free space on the right of L, which has a different background pattern, is not a part of any partition.

If you have more than one primary partition or logical drive on a disk, you can format them for different file systems. However, the format program will use the same sector size, number of tracks per cylinder, and number of sectors per track for each partition or logical drive. The cluster size can be different for each one.

The system partition is the partition on a disk that contains the hardware-specific files used in loading and initializing the operating system. Only a primary partition can be used as a system partition. Some programs, such as Disk Administrator and the MS-DOS-based program Fdisk, use the term active partition to refer to the same thing. This book uses the term system partition unless it is specifically describing dialog boxes or menus that use the term active or active partition.

The boot partition is also used in starting the operating system. It contains the operating system files and other support files needed by the operating system. Both a primary partition and a logical drive in an extended partition can be used as a boot partition. The system partition and the boot partition can be the same partition, or they can even be on different disks. There is more information about the system and boot partitions presented later in this chapter and in other chapters in "Reliability and Recoverability" part of the *Windows NT Server Resource Guide.*

Note To summarize, you can have as many as four primary partitions on a hard disk. If you have an extended partition on the disk, you can have up to three primary partitions. If the disk does not contain the system partition, you do not need to create any primary partitions. You can use the entire hard disk as an extended partition, and create as many logical drives within it as you want to have.

Windows NT requires that the system partition be a primary partition.

Some computers create an EISA configuration partition as the first primary partition on the hard disk. On these computers, you can only have three other primary partitions on the disk, or two primary partitions and one extended partition.

You create a volume set by combining multiple areas of free space on one or more hard disks into a single logical disk. When you create a volume set, Windows NT uses the HKEY_LOCAL_MACHINE\SYSTEM\DISK Registry subkey to access the areas on the disk(s) that are part of the volume set.

A volume set can be made from:

- Free space on multiple disks.
- Multiple areas of free space on one disk.
- Multiple areas of free space on multiple disks.

You use Disk Administrator to create and extend volume sets. See Disk Administrator Help for information about creating volume sets.

Each volume set can include up to 32 areas of free space from one or more physical disks. When creating a volume set, the free space can be an unallocated area within an extended partition, or an unpartitioned area elsewhere on the disk. Figure 3.4 shows a volume set E, which is made up of two areas in the extended partition on disk 0. Figure 3.4 also shows the two different types of free space. The free space between drives E and L is unallocated space in an extended partition. The free space at the right side of the screen shot is unpartitioned space.

You cannot use volume sets to avoid the limit of four Partition Table entries for a disk. For example, the Disk Administrator screen shot in Figure 3.4 shows three primary partitions (drives C, D, and L) and one extended partition. The free space between the second part of volume set E and the primary partition L is part of the extended partition. It can be used to create one or more logical drives in the extended partition. However, the free space to the right of primary partition L is wasted disk space, because there are already three primary partitions and an extended partition.

You can tell whether free space is unallocated space in an extended partition or is unpartitioned space, because the diagonal lines go in different directions for the two types of free space.

A stripe set, a mirror set, and a stripe set with parity are each composed of identically sized areas on multiple disks. Like volume sets, Windows NT must use the HKEY_LOCAL_MACHINE\SYSTEM\DISK Registry subkey to know how to access the disks. You use Disk Administrator to create these types of volumes. For a description of stripe sets, mirror sets, and stripe sets with parity, see Chapter 4, "Planning a Reliable Configuration." For information about creating and using mirror sets and stripe sets with parity, see Chapter 5, "Preparing for and Performing Recovery."

Volume is the general term that refers to all of the following entities that you can create and use on a computer running Windows NT Server.

- Primary partition.
- Logical drive in an extended partition.
- Volume set.
- Stripe set.
- Mirror set.
- Stripe set with parity.

A volume has a single drive letter assigned to it, and is formatted for use by a file system.

See the section titled "Maximum Sizes," presented later in this chapter, for information about the maximum size of volumes.

Which Programs to Use to Create and Format Volumes

This section provides information about how to create and format volumes. You can create and format a volume in one of the following ways:

- During Windows NT Setup, you can:
  - Create and format a new primary partition into which to install Windows NT, provided you have enough unpartitioned disk space for the Windows NT installation (approximately 160 MB for Windows NT Server on a RISC-based computer).
  - Format an existing, unformatted volume into which to install Windows NT.
  - When you choose to install Windows NT in an existing FAT volume, you can convert the volume from the FAT file system to the NTFS file system. Converting preserves the data on the volume.
- If you have Windows NT installed on your computer, you can use Disk Administrator to create and format the volume. You can also use Disk Administrator to reformat an existing volume that is already formatted for the FAT file system or the NTFS file system. Reformatting a volume erases all information about which files used to be on the volume. You can reformat the volume as FAT or NTFS, regardless of what file system it is currently formatted for. For information about Disk Administrator procedures, see the Disk Administrator Help.
- If you want to specify special options during the format, such as a cluster size to use for a FAT volume, you can run the Format program from the command prompt. To see the options for the Format program, enter **format /?** at the command prompt.

- When running Windows NT, you can use the Convert program to convert an existing FAT volume to an NTFS volume, which preserves the data on the volume. For information about using the Convert program, see Chapter 7, "Disk, File System, and Backup Utilities."
- When you have started MS-DOS, you can use the **fdisk** command to create a primary partition or logical drive, and use the **format** command to format the volume for the FAT file system. See MS-DOS Help for information about these commands.
- On Alpha AXP-based computers and MIPS-based computers, you can use the Arcinst program to create and format volumes.

Note Converting a volume from the FAT file system to the NTFS file system can take a long time, depending on the amount of fragmentation and size of the FAT volume. For example, converting a 1 GB volume can take up to several hours. The conversion process has to read all of the files to build the information that NTFS needs to have about the data on the volume. When using the Convert program or converting the volume in Windows NT Setup, no status information is displayed to indicate that the conversion is proceeding. Although it looks like the conversion process is hung, it might just be taking a long time. For a large volume, it might be a good idea to run the file system conversion process overnight.

It might be faster to back up the data on the FAT volume to tape or a remote computer, reformat the volume as NTFS, and then restore the data.

If you create a single primary partition that occupies the entire disk, you cannot change the way your disk is organized without deleting the primary partition (which deletes all of the information in it) and creating new, smaller primary partitions or an extended partition. You have to back up the information that you need to save to a backup tape or a disk on a remote computer, reinstall Windows NT, and restore the data to the new volumes that you have created.

If there is no unpartitioned space left on your disk, the **Create** and **Create Extended** options on the Disk Administrator **Partition** menu are not available.

When you use Disk Administrator to create volumes, you have to write the changes to the Partition Table before you can format the volume. You know that you need to write the changes if the **Format** command is not available (grayed) on the **Tools** menu. You can write the changes by using one of these methods:

- Clicking **Commit Changes Now** on the **Partition** menu. You can now click the volume, click the **Tools** menu, and click **Format**.
- Exiting Disk Administrator. If you have made changes to your disk configuration, Disk Administrator displays a dialog box that asks whether you want to save changes. Click **Yes**. You can now restart Disk Administrator, click the volume, click the **Tools** menu, and click **Format**.

If you have created a volume but not formatted it, Disk Administrator displays the volume as unformatted. You cannot access the volume in either My Computer or Windows NT Explorer until it has been formatted.

You cannot see an NTFS volume when you are running any operating system other than Windows NT. This means that your drive letters will probably be different when you are running Windows NT and other operating systems. For example, if you have two FAT volumes (C and E), and an NTFS volume (D), and you start the MS-DOS operating system, the second FAT volume will be labeled D. Because MS-DOS does not recognize the NTFS volume, the drive letter that was assigned to the NTFS volume when running Windows NT is used for the second FAT volume when running MS-DOS.

In Disk Administrator, you cannot delete the Windows NT boot partition when you are running Windows NT. You can delete the Windows NT boot partition if it is not the same partition as the system partition and you start another operating system that does not use the Windows NT boot partition as its boot partition.

Formatting Hard Disks and Floppy Disks

Before you can use a hard disk or a floppy disk, you must format the disk.

There are three steps involved in formatting a hard disk:

- Low-level formatting the disk.
- Creating primary partitions or logical drives in an extended partition.
- Doing a logical formatting of the primary partitions or logical drives.

Formatting a floppy disk is much simpler than formatting a hard disk, because there are fewer steps. You cannot create partitions on a floppy disk, so there are only two steps:

- Low-level formatting the disk.
- Doing a logical formatting of the disk.

When you format a floppy disk, the format program performs both a low-level format and a logical format, so there is really only one step. However, the low-level format on a floppy disk is not as thorough as on a hard disk.

The rest of this section discusses low-level formatting, creating partitions, and logical formatting.

Low-level Formatting

Each disk vendor provides the low-level format program to use for the disk. Consult your disk documentation or contact the vendor to find out how to do low-level formatting.

Each low-level format program determines the correct sector size to use, the number of tracks, and the number of sectors per track. This format program selects sector size based upon the disk and the information about the sectors in its circuitry. The program for almost all disks used in the United States uses a sector size of 512 bytes. The program also writes error correction and sector identification information for each sector onto the disk.

SCSI disks should always be low-level formatted when put on a new controller. The reason for this is that the translation in use on the disk varies from controller to controller, and can even vary between two identical controllers based upon the controller settings.

On x86-based computers, you must enable the BIOS on the SCSI controller if your system partition is on a SCSI disk. Depending upon your disk configuration, you might need to turn off translation in the SCSI controller. Also make sure that the CMOS has no entries for SCSI disks. For more information about the SCSI controller, see Chapter 5, "Preparing for and Performing Recovery."

There are some older IDE disks that you should not low-level format, because the factory formatting puts information on the disk that a low-level format erases. However, newer IDE disks, and all enhanced IDE disks (also known as EIDE) need to be low-level formatted. You must configure an IDE or EIDE disk in the CMOS before you can use it on the computer. See your hardware documentation for information about changing information in the CMOS.

If you have created and formatted volumes on your hard disk, but are getting errors from the Chkdsk program or other disk scan utilities about bad sectors on your disk, the only way to permanently eliminate them is to do a low-level format. Low-level formatting maps around the bad sectors. However, a low-level format erases all data on the disk. Therefore, be sure to back up any data on the disk before you low-level format it. You then need to create volumes (primary partitions or logical drives in an extended partition), and logically format them before you can restore the data to the disk.

Caution Do not low-level format IDE hard disks unless the manufacturer's literature describes that you should do so.

Creating Partitions on the Disk

When you create primary partitions or an extended partition on a disk, you logically divide it into one or more areas that can be formatted for use by a file system. The first partition on the disk (whether it is a primary partition or an extended partition) always starts at the outside of the disk, at cylinder 0, head 0, and sector 1. Because partition boundaries are always cylinder boundaries, the smallest partition that you can create on a disk consists of all of the tracks on a single cylinder.

When you have Windows NT installed on your computer, you should use Disk Administrator to create partitions. However, if you are setting up a new computer, you might not have any operating system installed. But you still need to create and format a partition into which to install Windows NT.

On x86-based computers, if you have an MS-DOS bootable floppy disk with the Fdisk program on it, you can start your computer from the MS-DOS floppy disk and create partitions by using Fdisk. On RISC-based computers, you can use the Arcinst program on the Windows NT Server CD to create partitions. Windows NT Setup displays an option to create a partition to use for Windows NT, provided that there is enough unpartitioned space on a hard disk.

Note On x86-based computers, you might not be able to access all your hard disk when using **fdisk**. If you have a disk that is larger than 4 gigabytes (GB), MS-DOS cannot see the space beyond 4 GB. Depending on the geometry, this limitation could even be 1 GB, because MS-DOS cannot access volumes that go beyond cylinder 1023. If you have more than 2 SCSI disks, MS-DOS generally does not see any disks after the first two, unless you load a SCSI device driver.

When you create the first partition on a disk (either a primary partition or an extended partition), the program that you use creates the Master Boot Record and writes it to the first sector on the disk (cylinder 0, head 0, sector 1). The Master Boot Record contains the Partition Table, which has information about each partition defined on the disk. When you make any changes to volumes on the disk, such as creating, deleting, or formatting them, the program that you are using updates the Partition Table.

You can configure the entire disk as one primary partition, and logically format it for one file system. Or you can create more than one primary partition, and use different file systems on them. You can also create an extended partition, and create one or more logical drives within the extended partition.

For more information about primary partitions, logical drives, and extended partitions, see "Logical Organization: Partitions and Volumes," presented earlier in this chapter. For more information about the Master Boot Record and Partition Table, see "Disk Sectors Critical to the Startup Process," presented later in this chapter.

Logical Formatting

Before you can write any data to a primary partition or logical drive, you must do a logical format; that is, format it for use by a file system. Logical formatting writes information needed by the file system onto the disk. The information includes:

- Partition Boot Sector.
- Changing the System ID field in the Partition Table (hard disk only).
- File system metadata, including information about available and unused space and location of files and folders.
- Damaged areas.

When you have Windows NT installed on your computer, you should use Disk Administrator to format the volumes.

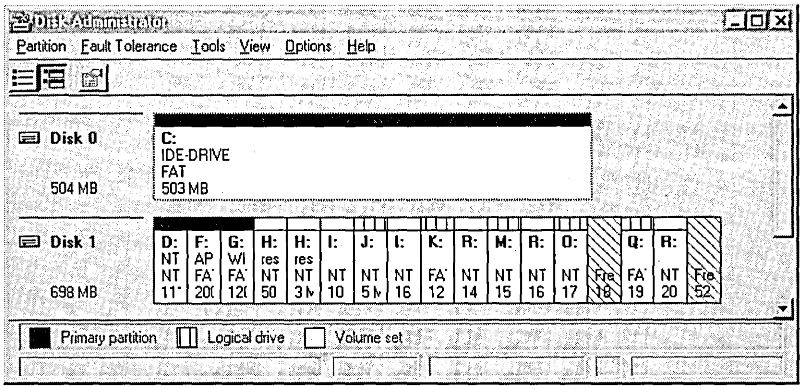
On x86-based computers, you can format primary partitions and logical drives for the FAT file system before you install any operating systems by starting the computer from a MS-DOS bootable floppy disk that has the Format program on it. You can use the Format program only for volumes that do not go beyond cylinder 1023.

On Alpha-based computers and MIPS-based computers, you can run the Arcinst program from the Windows NT Server product CD to format partitions on the hard disk.

On both x86-based computers and RISC-based computers, you can create and format a primary partition to use for Windows NT during Setup, provided that there is enough unpartitioned space on a hard disk.

Why Do You Need to Know Your Disk Configuration?

The next Disk Administrator screen shot is used to help you to understand the importance of knowing how the primary partitions and logical drives on your disks relate to each other, and why it is best if you can keep your disk configuration simple.



Note The key at the bottom of the Disk Administrator screen identifies the types of volumes that are configured on the computer. In the preceding screen shot, there are no stripe sets, mirror sets, or stripe sets with parity configured on the computer. Therefore, you do not see those types of volumes in the screen shot.

All that you can tell by looking at this screen shot is how the disks are configured and which volumes have been created. When using Disk Administrator, you cannot tell what a volume contains, or the order in which the members of a volume set, stripe set, or stripe set with parity were combined to form the volume.

▷ **To create a volume set, stripe set, or stripe set with parity**

1. Click an unpartitioned area on one of the disks.
2. Holding the CTRL key, click each of the other unpartitioned areas that you want to combine into the volume.
3. For a volume set or stripe set, click **Create Volume Set** or **Create Stripe Set** on the **Partition** menu. For a stripe set with parity, click **Create Stripe Set With Parity** on the **Fault Tolerance** menu.

The order in which you combined the areas on the disks into volume sets, stripe sets, or stripe sets with parity is critical information. It does not matter whether you first create a volume set, and then a stripe set, or the other way around. But it is absolutely critical to know in what order unpartitioned areas were added to a volume set that has been repeatedly extended.

It does not matter in what order you clicked the unpartitioned areas. What does make a difference is the order in which the disks were listed on the Disk Administrator screen when you selected them, and whether that order ever changes (for example, because you changed SCSI IDs). For example, the Disk Administrator screen shot at the beginning of this section has an IDE disk (disk 0) and a SCSI disk (disk 1). If you added a new SCSI disk between these two disks, and changed the SCSI ID of the current disk, you would not be able to access the volume sets on the disk that is currently disk 1.

The information that Windows NT uses to know how to access each of these volumes is in the HKEY_LOCAL_MACHINE\SYSTEM\DISK Registry subkey. The index to that Registry subkey is provided by a four-byte disk signature that is in the first sector of each hard disk. If this disk signature is ever changed, either because you replaced the disk, through corruption, or because someone modified it by using a low-level disk editor, you lose access to the volumes. You need to know the order in which the members of the volume were selected if you ever need to reconstruct the volume by using the FtEdit program (described in Chapter 7, "Disk, File System, and Backup Utilities").

To make matters even more complex, disks have different geometries. The geometry is the number of sectors per track, number of heads (or sides) on the disk, and the total number of cylinders on the disk.

The disk geometry is most important when you are configuring fault-tolerant volumes (mirror sets and stripe sets with parity). Having identical disk geometry is critical when you are configuring your system partition as a mirror set. For the most recoverable configuration, you should have identical primary partitions on both disks of the mirror set. Having different geometries means that you will never be able to start from your shadow disk without using the Windows NT startup floppy disk. Different geometries on the original and shadow system partition also mean that you can never start from your original system partition without using the Windows NT startup floppy disk when you have had to rebuild the disk (due to hardware failure or corruption). For details about mirroring the system partition, see the section titled "Configuring the System Partition on a Mirror Set," in Chapter 5, "Preparing for and Performing Recovery."

Now you have background information about why you need to know your disk configuration, and have an appreciation for the value of a simple configuration.

Disk Sectors Critical to the Startup Process

There are two sectors on the disk that are critical to starting the computer:

- Master Boot Record.
- Partition Boot Sector.

These sectors contain both executable code and the data required by the code. For more information about the startup process, see "What Happens When You Start Your Computer," presented later in this chapter. For a description of problems that you might encounter with these two sectors, see Chapter 6, "Troubleshooting Startup and Disk Problems."

Note Numbers larger than one byte are stored on x86-based computers in little endian format. For example, the sample value for the Relative Sector field in Table 3.1, 0x3F000000, is a little endian representation of 0x0000003F. The decimal equivalent of this little endian number is 63. Because most problems with the Master Boot Record and Partition Boot Sector occur on x86-based computers, the examples of these disk sectors are hexidecimal dumps from x86-based computers.

Master Boot Record

The Master Boot Record, created when you create the first partition on the hard disk, is probably the most important data structure on the disk. It is the first sector on the disk. The location is always track (cylinder) 0, head (side) 0, and sector 1.

The Master Boot Record contains the Partition Table for the disk and a small amount of executable code. On x86-based computers, the executable code examines the Partition Table, and identifies the system partition. The Master Boot Record then finds the system partition's starting location on the disk, and loads an copy of its Partition Boot Sector into memory. The Master Boot Record then transfers execution to executable code in the Partition Boot Sector.

Note Although there is a Master Boot Record on every hard disk, the executable code in the sector is used only if the disk is connected to an x86-based computer and the disk contains the system partition.

Figure 3.5 shows a hex dump of the sector containing the Master Boot Record. The figure shows the sector in two parts. The first part is the Master Boot Record, which occupies the first 446 bytes of the sector. The disk signature (FD 4E F2 14) is at the end of the Master Boot Record code. The second part is the Partition Table.

```
Physical Sector: Cyl 0, Side 0, Sector 1
00000000: 00 33 C0 8E D0 BC 00 7C - 8B F4 50 07 50 1F FB FC   .3.....|..P.P..
00000010: BF 00 06 B9 00 01 F2 A5 - EA 1D 06 00 00 BE BE 07   ................
00000020: B3 04 80 3C 80 74 0E 80 - 3C 00 75 1C 83 C6 10 FE   ...<.t..<.u.....
00000030: CB 75 EF CD 18 8B 14 8B - 4C 02 8B EE 83 C6 10 FE   .u......L.......
00000040: CB 74 1A 80 3C 00 74 F4 - BE 8B 06 AC 3C 00 74 0B   .t..<.t.....<.t.
00000050: 56 BB 07 00 B4 0E CD 10 - 5E EB F0 EB FE BF 05 00   V.......^.......
00000060: BB 00 7C B8 01 02 57 CD - 13 5F 73 0C 33 C0 CD 13   ..|...W.._s.3...
00000070: 4F 75 ED BE A3 06 EB D3 - BE C2 06 BF FE 7D 81 3D   Ou...........}.=
00000080: 55 AA 75 C7 8B F5 EA 00 - 7C 00 00 49 6E 76 61 6C   U.u.....|..Inval
00000090: 69 64 20 70 61 72 74 69 - 74 69 6F 6E 20 74 61 62   id partition tab
000000A0: 6C 65 00 45 72 72 6F 72 - 20 6C 6F 61 64 69 6E 67   le.Error loading
000000B0: 20 6F 70 65 72 61 74 69 - 6E 67 20 73 79 73 74 65    operating syste
000000C0: 6D 00 4D 69 73 73 69 6E - 67 20 6F 70 65 72 61 74   m.Missing operat
000000D0: 69 6E 67 20 73 79 73 74 - 65 6D 00 00 80 45 14 15   ing system...E..
000000E0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
000000F0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000100: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000110: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000120: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000130: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000140: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000150: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000160: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000170: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000180: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000190: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
000001A0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
000001B0: 00 00 00 00 00 00 00 00 - FD 4E F2 14 00 00         .........N....

                                                     80 01                  ..
000001C0: 01 00 06 0F 7F 96 3F 00 - 00 00 51 42 06 00 00 00   ....•.?...QB....
000001D0: 41 97 07 0F FF 2C 90 42 - 06 00 A0 3E 06 00 00 00   A....,.B...>....
000001E0: C1 2D 05 0F FF 92 30 81 - 0C 00 A0 91 01 00 00 00   .-....0.........
000001F0: C1 93 01 0F FF A6 D0 12 - 0E 00 C0 4E 00 00 55 AA   ...........N..U.
```

Figure 3.5. Hex Dump of the Master Boot Record for an x86-based computer

The disk signature is a unique number at offset 0x1B8 that Windows NT uses as an index to store and retrieve information about the disk in the Registry subkey HKEY_LOCAL_MACHINE\SYSTEM\DISK. The first time that you open Disk Administrator after formatting a hard disk, it displays a dialog box that informs the user that no disk signature was found on the disk. You should select **Yes**, or Windows NT will not be able to access the disk.

Many destructive viruses can damage the Master Boot Record and make it impossible to start the computer from the hard disk. Because the code in the Master Boot Record executes before any operating system is started, no operating system can detect or recover from corruption of the Master Boot Record. You can use the DiskProbe program on Windows NT Server Resource Kit CD to display the Master Boot Record, and compare it to the Master Boot Record shown in Figure 3.5. There are also utilities on the Resource Kit that enable you to save and restore the Master Boot Record. See Chapter 6, “Troubleshooting Disk and Startup Problems,” and Chapter 7, “Disk, File System, and Backup Utilities,” for more information.

Partition Table

The information about primary partitions and an extended partition is contained in the Partition Table, a 64-byte data structure located in the same sector as the Master Boot Record (cylinder 0, head 0, sector 1). The Partition Table conforms to a standard layout that is independent of the operating system. Each Partition Table entry is 16 bytes long, making a maximum of four entries available. Each entry starts at a predetermined offset from the beginning of the sector, as follows:

▪ Partition 1	0x01BE	(446)
▪ Partition 2	0x01CE	(462)
▪ Partition 3	0x01DE	(478)
▪ Partition 4	0x01EE	(494)

The last two bytes in the sector are a signature word for the sector and are always 0x55AA.

The next figure is a printout of the Partition Table for the disk shown in Figure 3.4. When there are fewer than four partitions, the remaining fields are all zeros.

```
                                                      80 01                ..
000001C0: 01 00 06 0F 7F 96 3F 00  - 00 00 51 42 06 00 00 00   ....•.?...QB....
000001D0: 41 97 07 0F FF 2C 90 42  - 06 00 A0 3E 06 00 00 00   A.....,.B...>....
000001E0: C1 2D 05 0F FF 92 30 81  - 0C 00 A0 91 01 00 00 00   .-....0.........
000001F0: C1 93 01 0F FF A6 D0 12  - 0E 00 C0 4E 00 00 55 AA   ...........N..U.
```

Figure 3.6. Partition Table from an x86-based computer

The following table describes each entry in the Partition Table. The sample values correspond to the information for partition 1 in Figure 3.6.

Table 3.1 Partition Table Fields

Byte Offset	Field Length	Sample Value	Meaning
00	BYTE	0x80	Boot Indicator. Indicates whether the partition is the system partition. Legal values are: 00 = Do not use for booting. 80 = System partition.
01	BYTE.	0x01	Starting Head.
02	6 bits	0x01	Starting Sector. Only bits 0-5 are used. Bits 6-7 are the upper two bits for the starting cylinder field.
03	10 bits	0x00	Starting Cylinder. This field contains the lower 8 bits of the cylinder value. Starting cylinder is thus a 10-bit number, with a maximum value of 1023.
04	BYTE	0x06	System ID. This byte defines the volume type. In Windows NT, it also indicates that a partition is part of a volume that requires the use of the HKEY_LOCAL_MACHINE\SYSTEM\DISK Registry subkey.
05	BYTE	0x0F	Ending Head.
06	6 bits	0x3F	Ending Sector. Only bits 0-5 are used. Bits 6-7 are the upper two bits for the ending cylinder field.
07	10 bits	0x196	Ending Cylinder. This field contains the lower 8 bits of the cylinder value. Ending cylinder is thus a 10-bit number, with a maximum value of 1023.
08	DWORD	3F 00 00 00	Relative Sector.
12	DWORD	51 42 06 00	Total Sectors.

The remainder of this section describes the uses of these fields. Definitions of the fields in the Partition Table is the same for primary partitions, extended partitions, and logical drives in extended partitions.

Boot Indicator Field

The Boot Indicator field indicates whether the volume is the system partition. On x86-based computers, only one partition on the disk should have this field set. This field is used only on x86-based computers. On RISC-based computers, the nonvolatile RAM (NVRAM) contains the information for finding the files to load.

System ID Field

For primary partitions and logical drives, the System ID field describes the file system used to format the volume. Windows NT uses this field to determine what file system device drivers to load during startup. It also identifies the extended partition, if there is one defined.

These are the values for the System ID field:

Value	Meaning
0x01	12-bit FAT partition or logical drive. The number of sectors in the volume is fewer than 32680.
0x04	16-bit FAT partition or logical drive. The number of sectors is between 32680 and 65535.
0x05	Extended partition. See section titled "Logical Drives and Extended Partitions," presented later in this chapter, for more information.
0x06	BIGDOS FAT partition or logical drive.
0x07	NTFS partition or logical drive.

Figure 3.6, presented earlier in this section, has examples of a BIGDOS FAT partition, an NTFS partition, an extended partition, and a 12-bit FAT partition.

When you create a volume set, stripe set, mirror set, or stripe set with parity, Disk Administrator sets the high bit of the System ID field for each primary partition or logical drive that is a member of the volume. For example, a FAT primary partition or logical drive that is a member of one of these volumes has a System ID value of 0x86. An NTFS primary partition or logical drive has a System ID value of 0x87. This bit indicates that Windows NT needs to use the HKEY_LOCAL_MACHINE\SYSTEM\DISK Registry subkey to determine how the members of the volume relate to each other. Volumes that have the high bit set can only be accessed by Windows NT. This bit is called the fault-tolerant (FT) bit.

When a primary partition or logical drive that is a member of one of these volumes has failed due to write errors or cannot be accessed, the second most significant bit is set. The System ID byte is set to C6 in the case of a FAT volume, or C7 in the case of an NTFS volume.

Note If you start up MS-DOS, it can only access primary partitions or logical drives that have a value of 0x01, 0x04, 0x5, or 0x6 for the System ID. However, you should be able to delete volumes that have the other values. If you use a MS-DOS-based low-level disk editor, you can read and write any sector, including ones that are in NTFS volumes.

Starting and Ending Head, Sector, and Cylinder Fields

On x86-based computers, the Starting and Ending Head, Cylinder, and Sector fields on the startup disk are very important for starting up the computer. The code in the Master Boot Record uses these fields to find and load the Partition Boot Sector.

The Ending Cylinder field in the Partition Table is ten bits long, which limits the maximum cylinder number that can be described in the Partition Table to 1023. The Starting and Ending Head fields are one byte long, which limits this field to the range 0 – 255. The Starting and Ending Sector field is 6 bits long, limiting its range to 0 – 63. However, sectors start counting at 1 (versus 0 for the other fields), so the maximum number of sectors per track is 63.

Since current hard disks are low-level formatted with the industry standard 512-byte sector size, the maximum capacity disk that can be described by the Partition Table can be calculated as follows:

MaxCapacity = (sector size) x (sectors per track) x (cylinders) x (heads)

Substituting the maximum possible values yields:

512 x 63 x 1024 x 256 = 8,455,716,864 bytes or 7.8 GB (GB)

This calculation results in a maximum formatted capacity is slightly less than 8 GB.

However, the maximum cluster size that you can use for FAT volumes when running Windows NT is 64K, when using a 512 byte sector size. Therefore, the maximum size for a FAT volume is 4 GB.

If you have a dual-boot configuration with Windows 95 or MS-DOS, FAT volumes that might be accessed when using either of those operating systems are limited to 2 GB. In addition, Macintosh computers that are viewing volumes on a computer running Windows NT cannot see more than 2 GB. If you try to use a FAT volume larger than 2 GB when running MS-DOS or Windows 95, or access it from a Macintosh computer, you might get a message that there are 0 bytes available.

The maximum size of a FAT volume on a specific computer depends on the disk geometry, and the maximum values that can fit in the fields described in this section. The next table shows the typical size of a FAT volume when translation is enabled, and when it is disabled. The number of cylinders in both situations is 1024.

Translation mode	Number of heads	Sectors per track	Maximum size for system or boot partition
Disabled	64	32	1 GB
Enabled	255	63	4 GB

Note RISC-based computers do not have a limit on the size of the system or boot partitions.

If a primary partition or logical drive extends beyond cylinder 1023, all of these fields will contain the maximum values.

Relative Sectors and Number of Sectors Fields

For primary partitions, the Relative Sectors field represents the offset from the beginning of the disk to the beginning of the partition, counting by sectors.The Number of Sectors field represents the total number of sectors in the partition. For a description of these fields in extended partitions, see the section "Logical Drives and Extended Partitions," presented later in this chapter.

Windows NT uses these fields to access all volumes. When you format a volume when running Windows NT, it puts data into the Starting and Ending Cylinder, Head, and Sector fields only for backward compatibility with MS-DOS and Windows 95, and to maintain compatibility with the BIOS interrupt (INT) 13 for startup purposes.

See the section titled "Using Disks With More Than 1024 Cylinders (x86-based Computers)," presented later in this chapter, for more information about the use of these fields.

Logical Drives and Extended Partitions

When more than four logical disks are required on a single physical disk, the first partition should be a primary partition. The second partition can be created as an extended partition, which can contain all the remaining unpartitioned space on the disk.

Note A primary partition is one that can be used as the system partition. If the disk does not contain a system partition, you can configure the entire disk as a single, extended partition.

Some computers create an EISA configuration partition as the first partition on the hard disk.

Windows NT detects an extended partition because the System ID byte in the Partition Table entry is set to 5. There can be only one extended partition on a hard disk.

Within the extended partition, you can create any number of logical drives. As a practical matter, the number of available drive letters is the limiting factor in the number of logical drives that you can define.

When you have an extended partition on the hard disk, the entry for that partition in the Partition Table (at the end of the Master Boot Record) points to the first disk sector in the extended partition. The first sector of each logical drive in an extended partition also has a Partition Table, which is the last 66 bytes of the sector. (The last two bytes of the sector are the end-of-sector marker.)

These are the entries in an extended Partition Table:

- The first entry is for the current logical drive.
- The second entry contains information about the next logical drive in the extended partition.
- Entries three and four are all zeroes.

This format repeats for every logical drive. The last logical drive has only its own entry listed. The entries for partitions 2-4 are all zeroes.

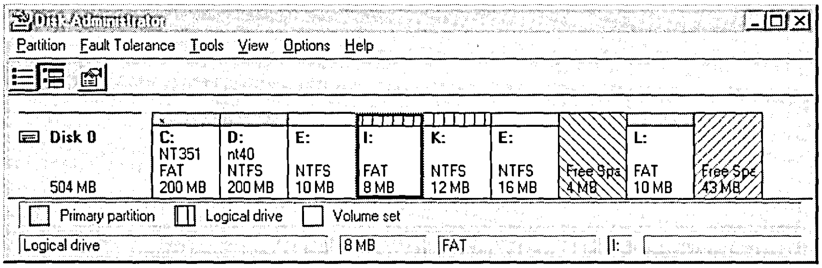
The Partition Table entry is the only information on the first side of the first cylinder of each logical drive in the extended partition. The entry for partition 1 in each Partition Table contains the starting address for data on the current logical drive. And the entry for partition 2 is the address of the sector that contains the Partition Table for the next logical drive.

The use of the Relative Sector and Total Sectors fields for logical drives in an extended partition is different than for primary partitions. For the partition 1 entry of each logical drive, the Relative Sectors field is the sector from the beginning of the logical drive that contains the Partition Boot Sector. The Total Sectors field is the number of sectors from the Partition Boot Sector to the end of the logical drive.

For the partition 2 entry, the Relative Sectors field is the offset from the beginning of the extended partition to the sector containing the Partition Table for the logical drive defined in the Partition 2 entry. The Total Sectors field is the total size of the logical drive defined in the Partition 2 entry.

Note If a logical drive is part of a volume set, the Partition Boot Sector is at the beginning of the first member of the volume set. Other members of the volume set have data where the Partition Boot Sector would normally be located.

This Disk Administrator screen shot shows a disk having an extended partition. The extended partition contains a volume set (drive E), two logical drives (drives I and K), and 4 MB of unallocated space. This screen shot is identical to the one shown in Figure 3.4.



Logical drive 8 MB FAT I:

The following example is a printout from the DiskMap program of the information in the Master Boot Record (the section titled MBR) and the extended Partition Table entries (the sections titled EBR) for the disk configuration shown in the preceding Disk Administrator screen shot. For information about DiskMap, see Chapter 7, "Disk, File System, and Backup Utilities."

```
MBR:
         Starting                   Ending        System    Relative   Total
  Cylinder Head Sector   Cylinder Head Sector      ID       Sector     Sectors
*        0    1      1        406   15     63     0x06          63      410193
       407    0      1        812   15     63     0x07      410256      409248
       813    0      1        914   15     63     0x05      819504      102816
       915    0      1        934   15     63     0x01      922320       20160

EBR: (sector 819504)
         Starting                   Ending        System    Relative   Total
  Cylinder Head Sector   Cylinder Head Sector      ID       Sector     Sectors
       813    1      1        832   15     63     0x87          63       20097
       833    0      1        848   15     63     0x05       20160       16128

EBR: (sector 839664)
         Starting                   Ending        System    Relative   Total
  Cylinder Head Sector   Cylinder Head Sector      ID       Sector     Sectors
       833    1      1        848   15     63     0x01          63       16065
       849    0      1        872   15     63     0x05       36288       24192

EBR: (sector 855792)
         Starting                   Ending        System    Relative   Total
  Cylinder Head Sector   Cylinder Head Sector      ID       Sector     Sectors
       849    1      1        872   15     63     0x07          63       24129
       873    0      1        905   15     63     0x05       60480       33264

EBR: (sector 879984)
         Starting                   Ending        System    Relative   Total
  Cylinder Head Sector   Cylinder Head Sector      ID       Sector     Sectors
       873    1      1        905   15     63     0x87          63       33201
         0    0      0          0    0      0     0x00           0           0
```

Partition Boot Sector

The Partition Boot Sector contains information that the file system uses to access the volume. On x86-based computers, the Master Boot Record use the Partition Boot Sector on the system partition to load the operating system Kernel files, or in the case of Windows NT, the boot loader.

The Windows NT Partition Boot Sector consists of:

- A jump instruction.
- The OEM name and version.
- A data structure called the BIOS Parameter Block.
- Another data structure called the extended BIOS Parameter Block.
- The bootstrap code.

The BIOS Parameter Block (bytes 11 through 35 of the Partition Boot Sector) describes physical parameters of the volume. The extended BIOS Parameter Block begins immediately after the BIOS Parameter Block. The length of this structure is different for FAT and NTFS volumes. The information in the BIOS Parameter Block and the extended BIOS Parameter Block is used by disk device drivers to read and configure volumes.

The Partition Boot Sector is the first 512-byte sector at the beginning of the volume. The sector always ends with 0x55AA, which is a sector signature word that marks the end of the sector.

FAT Partition Boot Sector

Table 3.2 describes the fields in the Partition Boot Sector for a volume formatted with the FAT file system. The sample values are the data in the corresponding fields in the hex dump in Figure 3.7, presented later in this section.

Table 3.2 Partition Boot Sector for FAT volumes

Byte Offset (in hex)	Field Length	Sample Value	Meaning
00	3 bytes	EB 3C 90	Jump instruction
03	8 bytes	MSDOS5.0	OEM Name in text
0B	25 bytes		BIOS Parameter Block
24	26 bytes		Extended BIOS Parameter Block
3E	448 bytes		Bootstrap code
1FE	2 bytes	0x55AA	End of sector marker

The following figure is a hex printout of the Partition Boot Sector for a FAT volume. The printout is formatted in three sections:

- Bytes 0 - 0x0A are the jump instruction and the OEM name.
- Bytes 0x0B - 0x3D are the BIOS Parameter Block and the Extended BIOS Parameter Block.
- The rest of the printout is the bootstrap code and the end of sector marker.

```
Physical Sector: Cyl 0, Side 1, Sector 1
00000000: EB 3C 90 4D 53 44 4F 53 - 35 2E 30                 <.MS-DOS5.0

                                             00 02 08 01 00              .....
00000010: 02 00 02 00 00 F8 C9 00 - 3F 00 10 00 3F 00 00 00   ........?...?...
00000020: 51 42 06 00 80 00 29 CE - 13 46 30 4E 4F 20 4E 41   QB....)..F0NO NA
00000030: 4D 45 20 20 20 20 46 41 - 54 31 36 20 20 20         ME    FAT16

                                                      33 C0                 3.
00000040: 8E D0 BC 00 7C 68 C0 07 - 1F A0 10 00 F7 26 16 00   ....|h......&..
00000050: 03 06 0E 00 50 91 B8 20 - 00 F7 26 11 00 8B 1E 0B   ....P.. ..&.....
00000060: 00 03 C3 48 F7 F3 03 C8 - 89 0E 08 02 68 00 10 07   ...H........h...
00000070: 33 DB 8F 06 13 02 89 1E - 15 02 0E E8 90 00 72 57   3.............rW
00000080: 33 DB 8B 0E 11 00 8B FB - 51 B9 0B 00 BE DC 01 F3   3.......Q.......
00000090: A6 59 74 05 83 C3 20 E2 - ED E3 37 26 8B 57 1A 52   .Yt... ...7&.W.R
000000A0: B8 01 00 68 00 20 07 33 - DB 0E E8 48 00 72 28 5B   ...h. .3...H.r([
000000B0: 8D 36 0B 00 8D 3E 0B 02 - 1E 8F 45 02 C7 05 F5 00   .6...>....E.....
000000C0: 1E 8F 45 06 C7 45 04 0E - 01 8A 16 24 00 EA 03 00   ..E..E.....$....
000000D0: 00 20 BE 86 01 EB 03 BE - A2 01 E8 09 00 BE C1 01   . ..............
000000E0: E8 03 00 FB EB FE AC 0A - C0 74 09 B4 0E BB 07 00   .........t......
000000F0: CD 10 EB F2 C3 50 4A 4A - A0 0D 00 32 E4 F7 E2 03   .....PJJ...2....
00000100: 06 08 02 83 D2 00 A3 13 - 02 89 16 15 02 58 A2 07   .............X..
00000110: 02 A1 13 02 8B 16 15 02 - 03 06 1C 00 13 16 1E 00   ................
00000120: F7 36 18 00 FE C2 88 16 - 06 02 33 D2 F7 36 1A 00   .6........3..6..
00000130: 88 16 25 00 A3 04 02 A1 - 18 00 2A 06 06 02 40 3A   ..%.......*...@:
00000140: 06 07 02 76 05 A0 07 02 - 32 E4 50 B4 02 8B 0E 04   ...v....2.P.....
00000150: 02 C0 E5 06 0A 2E 06 02 - 86 E9 8B 16 24 00 CD 13   ............$...
00000160: 0F 83 05 00 83 C4 02 F9 - CB 58 28 06 07 02 76 11   .........X(...v.
00000170: 01 06 13 02 83 16 15 02 - 00 F7 26 0B 00 03 D8 EB   ..........&.....
00000180: 90 A2 07 02 F8 CB 42 4F - 4F 54 3A 20 43 6F 75 6C   ......BOOT: Coul
00000190: 64 6E 27 74 20 66 69 6E - 64 20 4E 54 4C 44 52 0D   dn't find NTLDR.
000001A0: 0A 00 42 4F 4F 54 3A 20 - 49 2F 4F 20 65 72 72 6F   ..BOOT: I/O erro
000001B0: 72 20 72 65 61 64 69 6E - 67 20 64 69 73 6B 0D 0A   r reading disk..
000001C0: 00 50 6C 65 61 73 65 20 - 69 6E 73 65 72 74 20 61   .Please insert a
000001D0: 6E 6F 74 68 65 72 20 64 - 69 73 6B 00 4E 54 4C 44   nother disk.NTLD
000001E0: 52 20 20 20 20 20 20 00 - 00 00 00 00 00 00 00 00   R      .........
000001F0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 55 AA   ..............U.
```

Figure 3.7. Partition Boot Sector for a FAT volume on x86-based computer

Table 3.3 shows the layout of the BIOS Parameter Block and the extended BIOS Parameter Block. The sample values correspond to the data in Figure 3.7.

Table 3.3 BIOS Parameter Block and Extended BIOS Parameter Block Fields for FAT volumes

Byte Offset	Field Length	Value	Meaning
0x0B	WORD	0x0002	Bytes per Sector. The size of a hardware sector. For most disks in use in the United States, the value of this field is 512.
0x0D	BYTE	0x08	Sectors Per Cluster. The number of sectors in a cluster. The default cluster size for a volume depends on the volume size and the file system.
0x0E	WORD	0x0100	Reserved Sectors. The number of sectors from the Partition Boot Sector to the start of the first file allocation table, including the Partition Boot Sector. The minimum value is 1. If the value is greater than 1, it means that the bootstrap code is too long to fit completely in the Partition Boot Sector.
0x10	BYTE	0x02	Number of file allocation tables (FATs). The number of copies of the file allocation table on the volume. Typically, the value of this field is 2.
0x11	WORD	0x0002	Root Entries. The total number of file name entries that can be stored in the root folder of the volume.
0x13	WORD	0x0000	Small Sectors. The number of sectors on the volume if the number fits in 16 bits (65535). For volumes larger than 65536 sectors, this field has a value of 0 and the Large Sectors field is used instead.
0x15	BYTE	0xF8	Media Type. Provides information about the media being used. A value of 0xF8 indicates a hard disk.
0x16	WORD	0xC900	Sectors per file allocation table (FAT). Number of sectors occupied by each of the file allocation tables on the volume.
0x18	WORD	0x3F00	Sectors per Track. The apparent disk geometry in use when the disk was formatted.
0x1A	WORD	0x1000	Number of Heads. The apparent disk geometry in use when the disk was formatted.

Table 3.3 BIOS Parameter Block and Extended BIOS Parameter Block Fields for FAT volumes *(continued)*

Byte Offset	Field Length	Value	Meaning
0x1C	DWORD	3F 00 00 00	Hidden Sectors. Same as the Relative Sector field in the Partition Table.
0x20	DWORD	51 42 06 00	Large Sectors. If the Small Sectors field is zero, this field contains the total number of sectors in the volume. If Small Sectors is nonzero, this field contains zero.
0x24	BYTE	0x80	Physical Disk Number. This is related to the BIOS physical disk number. Floppy drives are numbered starting with 0x00 for the A disk. Physical hard disks are numbered starting with 0x80. The value is typically 0x80 for hard disks, regardless of how many physical disk drives exist, because the value is only relevant if the device is the startup disk.
0x25	BYTE	0x00	Current Head. Not used by the FAT file system. Therefore, Windows NT uses this field to store two flags: ▪ The low order bit is a dirty flag, used to indicate that Windows NT should run the Chkdsk program against the volume when it starts. ▪ The second lowest bit indicates to that a surface scan should also be run.
0x26	BYTE	0x29	Signature. Must be either 0x28 or 0x29 in order to be recognized by Windows NT.
0x27	4 bytes	CE 13 46 30	Volume Serial Number. A unique number that is created when you format the volume.
0x2B	11 bytes	NO NAME	Volume Label. This field was used to store the volume label, but the volume label is now stored as special file in the root directory.
0x36	8 bytes	FAT16	System ID. Either FAT12 or FAT16, depending on the format of the disk.

NTFS Partition Boot Sector

Table 3.4 describes the fields in the Partition Boot Sector for a volume formatted with the NTFS file system. The sample values are the data in the corresponding fields in the hex dump in Figure 3.8, presented later in this section.

Table 3.4 Partition Boot Sector for NTFS volume

Byte Offset (in hex)	Field Length	Sample Value	Meaning
00	3 bytes	EB 5B 00	Jump instruction
03	8 bytes	NTFS	OEM Identifier
0B	25 bytes		BIOS Parameter Block
24	48 bytes		Extended BIOS Parameter Block
54	426 bytes		Bootstrap code
1FE	WORD	0x55AA	End of sector marker

Note The bootstrap code for an NTFS volume is longer than the 426 bytes shown here. When you format an NTFS volume, the format program allocates the first 16 sectors for the Partition Boot Sector and the bootstrap code.

The extra data fields after the BIOS Parameter Block form an extended BIOS Parameter Block for NTFS. The data in these fields enable the boot loader to find the Master File Table (MFT) during the startup process. In the NTFS file system, the MFT is not located in a predefined sector as is the case for the file allocation table on a FAT volume. Thus, the MFT can be moved if there is a bad sector in its normal location. However, if these data are corrupt, the MFT cannot be located and Windows NT assumes the volume has not been formatted.

Note The MFT contains information about all of the files on an NTFS volume. See Chapter 17, "Disk and File System Basics," in the *Windows NT Workstation Resource Guide* for details about the MFT.

The following figure is a hex dump of the Partition Boot Sector for an NTFS volume (formatted when running Windows NT 4.0). The printout is formatted in three sections:

- Bytes 0 - 0x0A are the jump instruction and the OEM name.
- Bytes 0x0B - 0x53 are the BIOS Parameter Block and the Extended BIOS Parameter Block.
- The rest of the printout is the bootstrap code and the end of sector marker.

```
Physical Sector: Cyl 407, Side 0, Sector 1
00000000: EB 5B 90 4E 54 46 53 20 - 20 20 20                  .[.NTFS

                                             00 02 01 00 00             .....
00000010: 00 00 00 00 00 F8 00 00 - 3F 00 10 00 90 42 06 00  ........?....B..
00000020: 00 00 00 00 80 00 80 00 - A0 3E 06 00 00 00 00 00  .........>......
00000030: 10 00 00 00 00 00 00 00 - 51 1F 03 00 00 00 00 00  ........Q.......
00000040: 02 00 00 00 04 00 00 00 - 5B 1F DD 2C 4F DD 2C A2  ........[.,O.,.
00000050: 00 00 00 00                                        ....

                      00 00 00 00 - 00 00 00 00 00 FA 33 C0      ..........3.
00000060: 8E D0 BC 00 7C FB B8 C0 - 07 8E D8 C7 06 54 00 00  ....|........T..
00000070: 00 C7 06 56 00 00 00 C7 - 06 5B 00 10 00 B8 00 0D  ...V.....[......
00000080: 8E C0 2B DB E8 07 00 68 - 00 0D 68 56 02 CB 50 53  ..+....h..hV..PS
00000090: 51 52 06 66 A1 54 00 66 - 03 06 1C 00 66 33 D2 66  QR.f.T.f....f3.f
000000A0: 0F B7 0E 18 00 66 F7 F1 - FE C2 88 16 5A 00 66 8B  .....f......Z.f.
000000B0: D0 66 C1 EA 10 F7 36 1A - 00 88 16 25 00 A3 58 00  .f....6....%..X.
000000C0: A1 18 00 2A 06 5A 00 40 - 3B 06 5B 00 76 03 A1 5B  ...*.Z.@;.[.v..[
000000D0: 00 50 B4 02 8B 16 58 00 - B1 06 D2 E6 0A 36 5A 00  .P....X......6Z.
000000E0: 8B CA 86 E9 8A 36 25 00 - B2 80 CD 13 58 72 2A 01  .....6%.....Xr*.
000000F0: 06 54 00 83 16 56 00 00 - 29 06 5B 00 76 0B C1 E0  .T...V..).[.v...
00000100: 05 8C C2 03 D0 8E C2 EB - 8A 07 5A 59 5B 58 C3 BE  ..........ZY[X..
00000110: 59 01 EB 08 BE E3 01 EB - 03 BE 39 01 E8 09 00 BE  Y.........9.....
00000120: AD 01 E8 03 00 FB EB FE - AC 3C 00 74 09 B4 0E BB  .........<.t....
00000130: 07 00 CD 10 EB F2 C3 1D - 00 41 20 64 69 73 6B 20  .........A disk
00000140: 72 65 61 64 20 65 72 72 - 6F 72 20 6F 63 63 75 72  read error occur
00000150: 72 65 64 2E 0D 0A 00 29 - 00 41 20 6B 65 72 6E 65  red....).A kerne
00000160: 6C 20 66 69 6C 65 20 69 - 73 20 6D 69 73 73 69 6E  l file is missin
00000170: 67 20 66 72 6F 6D 20 74 - 68 65 20 64 69 73 6B 2E  g from the disk.
00000180: 0D 0A 00 25 00 41 20 6B - 65 72 6E 65 6C 20 66 69  ...%.A kernel fi
00000190: 6C 65 20 69 73 20 74 6F - 6F 20 64 69 73 63 6F 6E  le is too discon
000001A0: 74 69 67 75 6F 75 73 2E - 0D 0A 00 33 00 49 6E 73  tiguous....3.Ins
000001B0: 65 72 74 20 61 20 73 79 - 73 74 65 6D 20 64 69 73  ert a system dis
000001C0: 6B 65 74 74 65 20 61 6E - 64 20 72 65 73 74 61 72  kette and restar
000001D0: 74 0D 0A 74 68 65 20 73 - 79 73 74 65 6D 2E 0D 0A  t..the system...
000001E0: 00 17 00 5C 4E 54 4C 44 - 52 20 69 73 20 63 6F 6D  ...\NTLDR is com
000001F0: 70 72 65 73 73 65 64 2E - 0D 0A 00 00 00 00 55 AA  pressed.......U.
```

Figure 3.8. Partition Boot Sector for a NTFS volume on an x86-based computer

The next table describes the fields for the BIOS Parameter Block and the Extended BIOS Parameter Block. The fields starting at 0x0B, 0x0D, 0x15, 0x18, 0x1A, and 0x1C are the same as the corresponding fields for FAT volumes, as described in Table 3.3.

Table 3.5 BIOS Parameter Block and Extended BIOS Parameter Block Fields for NTFS volumes

Byte Offset	Field Length	Sample Value	Meaning
0x0B	WORD	0x0002	Bytes per Sector.
0x0D	BYTE	0x01	Sectors per Cluster.
0x0E	WORD	0x0000	Reserved sectors.
0x10	BYTE	0x00	Always 0.
0x11	WORD	0x0000	Always 0.
0x13	WORD	0x0000	Not used by NTFS.
0x15	BYTE	0xF8	Media Type.
0x16	WORD	0x0000	Always 0.
0x18	WORD	0x3F00	Sectors per Track.
0x1A	WORD	0x1000	Number of Heads.
0x1C	DWORD	90 42 06 00	Hidden Sectors.
0x20	DWORD	00 00 00 00	Not used by NTFS.
0x24	DWORD	80 00 80 00	Not used by NTFS.
0x28	LONGLONG	A0 3E 06 00 00 00 00 00	Total Sectors.
0x30	LCN	10 00 00 00 00 00 00 00	Logical cluster number for $MFT.
0x38	LCN	51 1F 03 00 00 00 00 00	Logical cluster number for $MFTMirr.
0x40	DWORD	02 00 00 00	Clusters per file record segment.
0x44	DWORD	04 00 00 00	Clusters per index block.
0x48	LONGLONG	5B 1F DD 2C 4F DD 2C A2	Volume Serial Number.
0x50	DWORD	00 00 00 00	Checksum.

For information about the Master File Table and the $MFT and $MFTMirr files, see Chapter 17, "Disk and File System Basics," in the *Windows NT Workstation Resource Guide*.

What Happens When You Start Your Computer

This section describes what happens when you start a computer that has a Windows NT installed as one of the operating systems. In general, this section describes what happens at each step in the process when the computer successfully starts. Chapter 6, "Troubleshooting Startup and Disk Problems," discusses what you can do to isolate startup problems and how to recover from them.

These are the steps involved in the successful startup of Windows NT:

- Power On Self Test (POST) processing
- Initial startup process
- Boot loader process
- Selecting the operating system
- Detecting hardware
- Selecting a configuration
- Loading the Kernel
- Initializing the Kernel
- Logging on

The order and the processing is somewhat different, depending upon the hardware platform (x86-based computer or RISC-based computer).

The startup process begins when you:

- Power on the computer.
- Click **Shutdown and Restart** on the **Shutdown Computer** dialog box.
- Select **Restart the computer** from the **Shut Down Windows** dialog box.

When you see the **Begin Logon** dialog box, with the text **Press Ctrl+Alt+Delete to log on**, your computer has completed the loading of Windows NT, and has completed much of the initialization. However, startup is complete only when a user can successfully log on at the **Begin Logon** dialog box.

Windows NT might not start up or operate correctly if any hardware components do not initialize correctly. Startup fails if any of the files required to start Windows NT are not present in the correct folder, or if one of the files has been corrupted. Table 3.6 describes the files that x86-based computers and RISC-based computers use to start Windows NT.

Table 3.6 Files required to start up Windows NT

x86-based file	Folder	RISC-based file	Folder
NTLDR	root of startup disk	Osloader.exe	\Os\Nt40
Boot.ini	root of startup disk	no equivalent	
Bootsect.dos	root of startup disk	no equivalent	
Ntdetect.com	root of startup disk	no equivalent	
Ntbootdd.sys (SCSI only)	root of startup disk		
Ntoskrnl.exe	*%systemroot%*\ System32	Ntoskrnl.exe	*%systemroot%*\ System32
Hal.dll	*%systemroot%*\ System32	Hal.dll	\Os\Nt40
no equivalent		*.pal files (Alpha-based computers)	\Os\Nt40
SYSTEM key	*%systemroot%*\ System32\Config	SYSTEM key	*%systemroot%*\ System32\Config
device drivers	*%systemroot%*\ System32\Drivers	device drivers	*%systemroot%*\ System32\Drivers

Power On Self Test Processing

When you power on or restart a computer, it goes through its Power On Self Test (POST) routine, which determines:

- The amount of real memory.
- Whether the needed hardware components (such as the keyboard) are present.

Once the computer has run its POST routine, each adapter card with a BIOS runs its own POST routine. The computer and adapter card manufacturers determine what appears on the screen during the POST processing.

Initial Startup Process

The first sector on the hard disk is critical to the startup process. This sector contains the Master Boot Record and the Partition Table.

If the startup disk is a floppy disk, the first sector on the disk is the Partition Boot Sector.

For more information about the Master Boot Record, the system partition, and the Partition Boot Sector, see "Disk Sectors Critical to the Startup Process," presented earlier in this chapter.

Starting an x86-based Computer

After the POST on x86-based computers, the system BIOS attempts to locate the startup disk. If there is a floppy disk in drive A, the system BIOS uses drive A as the startup disk. If there is no disk in drive A, it then checks the first hard disk that is powered up.

Note Some system BIOS versions enable the user to reconfigure the order in which it checks the floppy disks and hard disks for the startup disk.

When the hard disk is the startup disk, the system BIOS reads the Master Boot Record, and loads it into memory. The system BIOS then transfers execution to the Master Boot Record. The code in the Master Boot Record scans the Partition Table for the system partition. When the Master Boot Record finds the system partition, it loads sector 0 of the partition into memory, and executes the code in the sector. Sector 0 on the system partition can be a program or diagnostic program, or a Partition Boot Sector that contains startup code for an operating system. The Partition Boot Sector code starts the operating system in a manner defined by the operating system.

If there is no system partition on the first hard disk, the Master Boot Record displays errors such as the following:

- Invalid partition table
- Error loading operating system
- Missing operating system

If there is a floppy disk in drive A, the system BIOS loads the first sector on the disk into memory. If the floppy is bootable, the first sector on the disk is the Partition Boot Sector. If the floppy disk is not bootable, you see errors such as:

```
Non-System disk or disk error
Replace and press any key when ready
```

Starting a RISC-based Computer

After a RISC-based computer completes the POST routine, the resident ROM firmware selects a startup disk by reading a boot precedence table from nonvolatile RAM (NVRAM). What the firmware does depends upon the platform (Alpha, PPC, or MIPS computer) and the information in the NVRAM. For example, to startup from the floppy disk, the NVRAM must define it as an alternate boot selection.

The NVRAM also defines the:

- Location for OSLOADER.
- Path to the boot partition.
- Folder that contains the operating system to start.

The system partition can be on any hard disk on RISC-based computers. You can use the **Boot selection** menu to set up or change the system partition. See "Manage Boot Selection Menu," in Chapter 19, "What Happens When You Start Your Computer," of the *Windows NT Workstation Resource Guide* for more information about changing the NVRAM on RISC-based computers.

The system partition on a RISC-based computer must be formatted with the FAT file system.

Boot Loader Process

The boot loader enables you to select the operating system that you want to start and loads the operating system files from the boot partition. Boot loader processing is different on x86-based and RISC-based computers.

NTLDR Functions (x86-based Computers)

NTLDR controls the operating system selection process and hardware detection prior to the Windows NT Kernel initialization. NTLDR must be in the root folder.

When NTLDR starts executing, it clears the screen and displays the boot loader message, which is:

```
OS Loader V4.0
```

OSLOADER Functions (RISC-based Computers)

On RISC-based computers, Osloader.exe performs all of the functions that are performed by the x86-based components NTLDR, Ntdetect.com, and Bootsect.dos.

The NVRAM contains environment variables that provide the functional equivalent of the Boot.ini file on x86-based computers.

Selecting the Operating System to Start

The boot loader displays the screen from which you select the operating system to start. The information on the screen is different for x86-based computers and RISC-based computers.

Boot Loader Screen (x86-based Computer)

NTLDR displays a menu from which you select the operating system to start. This screen is based upon the information in the Boot.ini file. The screen looks like this:

```
OS Loader V4.0

Please select the operating system to start:

     Windows NT Server Version 4.0
     Windows NT Server Version 4.0 (VGA mode)

Use ↑ and ↓ to move the highlight to your choice.
Press Enter to choose.

Seconds until highlighted choice will be started automatically: 29
```

If you do not select an entry before the counter reaches zero, NTLDR loads the operating system specified by the default parameter in the Boot.ini file. Windows NT Setup sets the default entry to the most recent copy of Windows NT that you installed. You can edit the Boot.ini file to change the default entry if you want to default to an operating system other than the most recently installed version of Windows NT.

For information about changing the Boot.ini file, see "Contents and Purpose of Boot.ini," presented later in this chapter.

Boot Menu (RISC-based Computers)

The **Boot** menu contains the boot options. The first selection is always the default operating system, which is the most recently installed version of Windows NT unless you change the order of the boot selections.

To change the default operating system, or add or change alternate operating system choices, see the section titled "Manage Boot Selection Menu," in Chapter 19, "What Happens When You Start Your Computer," of the *Windows NT Workstation Resource Guide.*

This is a sample **Boot** menu from an Alpha-based computer:

```
ARC Multiboot Alpha AXP Version 3.5-11
Copyright (c) 1993 Microsoft Corporation
Copyright (c) 1993 Digital Equipment Corporation

Boot Menu:

    Boot Windows NT Server Version 4.0
    Boot an alternate operating system
    Run a program
    Supplementary menu. . .

Use the arrow keys to select, then press Enter.
Seconds until auto-boot. Select another option to override: 9
```

The first line on the **Boot** menu is highlighted. Pressing ENTER begins startup of the default operating system.

If you have included alternate boot selections in your NVRAM, you can select **Boot an alternate operating system**, which results in a screen like the following.

```
ARC Multiboot Alpha AXP Version 3.5-11
Copyright (c) 1993 Microsoft Corporation
Copyright (c) 1993 Digital Equipment Corporation

Boot Menu:

    Boot Windows NT Server 4.0 (Default)
    Boot Windows NT Server 3.51

Use the arrow keys to select, then press Enter.
```

Detecting Hardware

On RISC-based computers, Osloader.exe obtains the hardware information from the NVRAM.

Ntdetect.com is the hardware recognizer for x86-based computers. It collects a list of currently installed components and returns this information to NTLDR. Ntdetect.com executes after you select a Windows NT operating system on the boot loader screen (or the timer times out). When Ntdetect.com begins to execute, you see the following line on the screen:

```
NTDETECT V1.0 Checking Hardware . . .
```

Ntdetect.com detects the following components:

- Computer ID
- Bus/Adapter Type
- Video
- Keyboard
- Communication Ports
- Parallel Ports
- Floppy Disks
- Mouse/Pointing Device

Selecting a Configuration

When you have selected the version of Windows NT to start, and the boot loader has collected hardware information, you see the following screen:

```
OS Loader V4.0

Press spacebar now to invoke Hardware Profile/Last Known Good menu.
```

The boot loader waits a few seconds for you to press the spacebar. If you do not press the spacebar and you have only one hardware profile, the boot loader loads Windows NT by using the Default configuration. Otherwise, you see this screen:

```
    Hardware Profile/Configuration Recovery Menu

This menu allows you to select a hardware profile
to be used when Windows NT is started.

If your system is not starting correctly, then you may switch to a
previous system configuration, which may overcome startup problems.
IMPORTANT: System configuration changes made since the last successful
startup will be discarded.

    Original Configuration
    some other hardware profile

Use the up and down arrow keys to move the highlight
to the selection you want. Then press ENTER.
To switch to the Last Known Good Configuration, press ‘L’.
To Exit this menu and restart your computer. press F3.

Seconds until highlighted choice will be started automatically: 5
```

The first hardware profile is highlighted. If you have created other hardware profiles, use the down arrow to select the one that you want to use. For more information about hardware profiles, use the **Find** tab in Windows NT Help and enter **profile**.

You can also select between the Last Known Good Configuration and the Default Configuration. Windows NT automatically uses the Default Configuration if you do not select the Last Known Good Configuration. When you use the Default Configuration, the boot loader uses the Registry information that Windows NT saved at the last shutdown.

If you switch to the Last Known Good Configuration by pressing L and ENTER, the boot loader uses the Registry information that it saved at the completion of the last successful startup to configure this startup.

For more information about control sets, the Last Known Good Configuration, and the Default Configuration, see Chapter 8, "General Troubleshooting."

Loading the Kernel

When you press ENTER on the **Hardware Profile/Configuration Recovery** menu or the boot loader automatically makes the selection for you, the computer is in the Kernel load phase of Windows NT startup. You see several dots as the boot loader loads the Windows NT Kernel (Ntoskrnl.exe) and the hardware abstraction layer (Hal.dll) into memory.

At this point, the boot loader creates the control set it will use to initialize the computer. The control set is either the Default control set or the Last Known Good control set. The boot loader loads the low-level device drivers based on the ones identified in the control set.

Initializing the Kernel

You know that the Kernel is initializing when the screen turns blue, and you see text similar to the following:

```
Microsoft (R) Windows NT (TM) Version 4.0 (Build 1345)
1 System Processor (16 MB Memory)
```

This means that Ntoskrnl.exe has successfully initialized and that control has passed to it.

During the Kernel initialization phase, the Kernel:

- Initializes the low-level device drivers that were loaded in the previous phase.
- Loads and initializes the other device drivers.
- Runs programs, such as Chkdsk, that need to be run before it loads any services.
- Load and initializes services.
- Creates the page file.
- Starts subsystems that are needed to run Windows NT.

Logging on

When Kernel initialization finishes, you see the **Begin Logon** dialog box, which contains the text **Press Ctrl + Alt + Delete to log on.** At this time, Windows NT might still be initializing network drivers, but you can logon now.

Note Windows NT startup is not considered good until a user successfully logs on to the system.

Contents and Purpose of Boot.ini (x86-based Computers)

When you install Windows NT on an x86-based computer, Windows NT Setup puts the Boot.ini file at the root of the system partition. NTLDR uses information in the Boot.ini file to display the screen from which you select the operating system to start.

Here is a sample Boot.ini file:

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINNT
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Windows NT Server Ver 4.0"
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Windows NT Server Ver 4.0
[VGA mode]" /basevideo /sos
```

The Boot.ini file has two sections, [boot loader] and [operating system], which are described in the next two sections.

[boot loader] Section

This section contains the following information:

Parameter	Description
timeout	The number of seconds the user has to select an operating system from the boot loader screen before NTLDR loads the default operating system. If the value is 0, NTLDR immediately starts the default operating system without displaying the boot loader screen. You can also set this value to -1, in which case NTLDR waits forever for you to make a selection. You must edit the Boot.ini file to set the value to -1 because it is an illegal value for the System option in Control Panel.
default	The path to the default operating system.

[operating systems] Section

This section contains the list of available operating systems. Each entry includes the path to the boot partition for the operating system, the string to display in the boot loader screen, and optional parameters. There is an example of the use of this section of the Boot.ini file in the section titled "Boot Loader Screen (x86-based Computers)," presented earlier in this chapter.

The Boot.ini file supports starting multiple versions of Windows NT operating systems, as well as starting one other operating system. The other operating systems that can be included in the Boot.ini file are Windows 95, MS-DOS, and OS/2. The section "Configuring the Computer for Dual-booting and Triple-booting," in Chapter 19, "What Happens When You Start Your Computer," of the *Windows NT Workstation Resource Guide*, contains more information about the other operating systems.

Boot.ini Switches

There are several switches that you can add to the end of the Windows NT entries in the [operating system] section of the Boot.ini file. They are not case sensitive. Chapter 6, "Troubleshooting Startup and Disk Problems," contains more information about many of these switches. For information about the debugger, see Chapter 39, "Windows NT Debugger," in the *Windows NT Workstation Resource Guide.*

Table 3.7 Boot.ini Switches

Switch	Description
/BASEVIDEO	The computer starts up using the standard VGA video driver. If you have installed a new video driver, but it is not working correctly, selecting the Windows NT entry with this switch enables you to start the computer so you can change to a different driver.
/BAUDRATE	Specifies the baud rate to be used for debugging. If you do not set the baud rate, the default baud rate is 9600 if a modem is attached, and 19200 for a null-modem cable. This switch has a secondary effect of forcing the /DEBUG switch as well, whether or not use you use /DEBUG.
/CRASHDEBUG	The debugger is loaded when you start Windows NT, but remains inactive unless a Kernel error occurs. This mode is useful if you are experiencing random, unpredictable Kernel errors.
/DEBUG	The debugger is loaded when you start Windows NT, and can be activated at any time by a host debugger connected to the computer. This is the mode to use when you are debugging problems that are regularly reproducible.
/DEBUGPORT= com*x*	Specifies the com port to use for debugging, where *x* is the communications port that you want to use. Like /BAUDRATE, this switch will also force the /DEBUG mode.
/MAXMEM:n	Specifies the maximum amount of RAM that Windows NT can use. This switch is useful if you suspect a memory chip is bad. See Chapter 6, “Troubleshooting Disk and Startup Problems,” for more information.
/NODEBUG	No debugging information is being used.
/NOSERIALMICE =[COM*x* | COM*x,y,z*...]	Disables serial mouse detection of the specified COM port(s). Use this switch if you have a component other than a mouse attached to a serial port during the startup sequence. If you use /NOSERIALMICE without specifying a COM port, serial mouse detection is disabled on all COM ports. See Chapter 5, “Preparing for and Performing Recovery,” for more information.
/SOS	Displays the driver names while they are being loaded. Use this switch if Windows NT won’t start up and you think a driver is missing. See Chapter 6, “Troubleshooting Startup and Disk Problems,” for more information.

Editing Boot.ini

When you install Windows NT, the Boot.ini file has the Read Only, System, and Hidden attributes set. You can edit the timeout and default parameters in the Boot.ini file by using the System option on Control Panel, regardless of the value of these attributes.

If you want to edit the Boot.ini file by using a text editor, you need to make the file visible before you can open it, and you need to turn off the Read Only attribute to be able to make changes to it. You can change the attributes by using My Computer, Windows NT Explorer, or the command prompt.

▷ **To change the attributes in My Computer or Windows NT Explorer**

1. On the **View** menu, click **Options**.
2. On the **Options** tab, select **Show all files**. Click **OK**.
3. Click the filename.
4. On the **File** menu, click **Properties**.
5. In the **Attributes** box of the **General** tab, clear the **Read Only**, **System**, and **Hidden** check boxes, and then click **OK**.

To change the attributes by using command prompt, enter:

```
attrib -s -h -r boot.ini
```

If you change the path to the Windows NT boot partition, make sure to edit both the default path and operating system path entries. If you change one but not the other, a new choice is added to the boot loader screen, with the default designator next to it.

Using Hard Disks With More Than 1024 Cylinders (x86-based Computers)

Windows NT can take advantage of hard disks with more than 1024 cylinders if you enable extended translation. Windows NT does not have a limit to the number of cylinders it can address, but the system and boot partitions are limited to 1024 cylinders by the system BIOS, which must use INT 13 to access the disk during startup. The section "Starting and Ending Head, Sector, and Cylinder Fields," presented earlier in this chapter, contains more information about the system and boot partitions.

The following error messages might appear when extended translation is not enabled on disks larger than 1024 cylinders, and the Master Boot Record is unable to locate the Partition Boot Sector:

- Error loading operating system.
- Missing operating system.

If you create your system and boot partitions in Windows NT Setup, you do not have to concern yourself with these details, because Setup will not create a partition that cannot be used to start Windows NT.

When Windows NT has completed startup, it does not use the values in the Starting and Ending Head, Sector, and Cylinder fields of the Partition Table. Instead, it uses the Relative Sectors and Number Of Sectors fields. These fields provide a full 32 bits to represent sectors, which results in volumes of up to 2^{32} sectors.

With a standard sector size of 512 bytes, the 32-bit fields used for the Relative Sectors and Number of Sectors result in a maximum possible volume size of:

512 bytes x 4,294,967,296 = 2,199,023,255,552 bytes, or 2 TB.

When creating primary partitions or logical drives on a disk, Windows NT writes the correct values to the Partition Table fields whenever possible. When the Ending Cylinder field for an entry in the Partition Table goes beyond cylinder 1023, Windows NT writes the maximum permitted values to these fields in the Partition Table. This prevents the system BIOS from calculating the Starting and Ending addresses based on erroneous data.

There are several Knowledge Base articles that discuss using large disks. Knowledge Base articles are included on the *Windows NT Server Resource Kit.*

SCSI Hard Disks

If your SCSI controller can be configured to perform translation, you can create a boot partition or system partition that is within the first 1 or 4 GB of the disk, depending upon the translation used by the controller. To create the largest possible system and boot partitions, you need to enable translation on the controller. If translation was not already enabled, you will need to low-level format the hard disk, recreate partitions, and format them. Check with your hardware manufacturer for specific advice and details.

Once Windows NT has been initialized, it uses its SCSI disk device drivers to directly interact with the disk without using the BIOS INT 13 interface. Therefore, the Starting and Ending Head, Sector, and Cylinder fields are not important after the computer has completed startup.

Note If you are using, or going to use, a mirror set for your system partition, there are reasons that you should not enable translation on the controller. See Chapter 5, "Preparing for and Performing Recovery," for more information.

IDE and EIDI Hard Disks

The system and boot partitions on IDE hard disks are also limited to 1024 cylinders if the controller does not translate the disk address. Secondary disks that do not contain the system or boot partition might not be limited to 1024 cylinders, depending on the translation scheme used by the controller and disk.

IDE disks use a different data structure for representing the number of cylinders, number of heads, and number of sectors per track than the Partition Table and BIOS INT 13 interface. According to the IDE specifications, the maximum number of cylinders is 65536, the maximum number of heads is 16, and the maximum number of sectors per track is 255. The maximum size is 136.9 GB. Because the BIOS/IDE interface has been constrained to use the lowest common denominator, the result is a limit of about 504 MB for an IDE disk on an Intel-based ISA bus computer.

There are new controllers and disk device drivers that use an Enhanced Disk Parameter Table to translate between the logical sector layout internal to the IDE disk and the logical sector layout addressable by the BIOS INT 13 interface. These disk drivers and devices can support larger disks. These disks and controllers are referred to as enhanced IDE, or EIDE, devices.

ESDI Hard Disks

ESDI disks are an older type that came before the IDE disks.

ESDI disks use the CHS (cylinder, head, sector) addressing and have a 1024 cylinder limitation. Some controller cards get around this limitation by implementing a translation scheme in the onboard controller BIOS.

The translation is totally transparent to Windows NT, and the standard disk device driver works without any change.

ESDI disks are capable of being low-level formatted with various values of sector per track, such as 53 or 63. Here are two examples:

1024 cylinders x 15 heads x 53 sectors per track x 512 bytes per sector = 398 MB

1024 cylinders x 15 heads x 63 sectors per track x 512 bytes per sector = 472 MB

Thus, using 63 sectors per track yields 74 MB more space. Windows NT is compatible with either geometry, and, depending on the disk or controller, can access cylinders beyond 1023. You can use Windows NT to create and format the disk space beyond cylinder 1023, but it cannot be accessed by MS-DOS. However, some controllers can remap the remaining cylinders beyond 1023 so that either MS-DOS or Windows NT can use the entire capacity. For example:

1632 cylinders x 15 heads x 53 sectors per track x 512 bytes per sector = 634 MB

Microsoft has tested the ESDI controllers described in the following table.

Table 3.8 Using More Than 1024 Cylinders on ESDI Controllers

Controller	Achieving Maximum Capacity
Data Technology Corp. (DTC) Model 6282-24	The maximum MS-DOS geometry is 63 sectors per track and 1024 cylinders. Windows NT can access the cylinders beyond 1023. Do not perform a low-level format on the disk using Head Mapping mode.
Data Technology Corp. (DTC) Model 6290-24	The maximum MS-DOS geometry is 63 sectors per track and 1024 cylinders. Because this controller card does not have an on-board BIOS, Windows NT cannot access the cylinders beyond 1023.
Data Technology Corp. (DTC) Model 6290 SEZ (Dual SCSI/ESDI Controller)	The maximum MS-DOS geometry is 63 sectors per track and 1024 cylinders. Windows NT can access the cylinders beyond 1023.
Data Technology Corp. (DTC) Model 6295-24	The maximum MS-DOS geometry is 63 sectors per track and 1024 cylinders. This controller card does have an on-board BIOS, so Windows NT can access the cylinders beyond 1024. Do not perform a low-level format on the disk using Head Mapping mode.
Adaptec Model 2322D	Option 1. Disable disk translation and the on-board controller BIOS. In the CMOS, use a user-defined disk type with the actual disk parameters, such as Disk Type=48, Cylinders=1632, Heads=15, Sectors per Track=53. These parameters result in a 634 MB capacity. MS-DOS is limited to the first 1024 cylinders, which makes 398 MB available. Windows NT can access beyond cylinder 1023, yielding another 236 MB. Option 2. Both MS-DOS and Windows NT can access the entire disk. Set jumpers on the controller for Disk Splitting. Disable disk translation and the on-board controller BIOS. Set up the first physical disk in CMOS as Cylinders=1024, Heads=15, Sectors per Track=53. This gives a disk capacity of 398 MB. Set up the second disk (it appears as a physical disk) in the CMOS as Cylinders=606, Heads=15, sectors per track=53. This yields another 236 MB. Windows NT sees the disk as two physical disks.
UltraStor with PROM versions < nnnn-009	The maximum MS-DOS geometry is 63 sectors per track and 1024 cylinders. Windows NT can access the cylinders beyond 1023.
UltraStor with PROM versions >= nnnn-009	The maximum MS-DOS geometry is 63 sectors per track and 1024 cylinders. Both MS-DOS and Windows NT have access to the entire disk capacity.

Using Removable Disks and Floppy Disks

Most types of removable media can work with Window NT. Even though removable media are assigned a floppy disk icon in My Computer and Windows NT Explorer, they are not handled by the same part of the operating system. Floppy disks are handled by Floppy.sys. All SCSI-based removable media are handled through the normal SCSI disk device drivers. Most magneto-optical, Bernoulli, Syquest, and similar removable-media disks are treated like hard disks under Windows NT, and do not require special disk device drivers.

You can use My Computer or Windows NT Explorer to remove media while Windows NT is running. When pointing at the device, click the right mouse button. If the media can be ejected, the **Eject** command is available. If there is no **Eject** command in the menu, or the command is grayed, you cannot eject the media.

Note WORM and other types of "Write once" media are not supported in Windows NT.

There are several limitations that are unique to removable media in Windows NT.

- Removable media can only have a single primary partition. Extended partitions are not supported.
- Removable media devices cannot be used as part of a volume set, stripe set, mirror set, or stripe set with parity.
- It is not possible to format a floppy disk with the NTFS file system. The Windows NT operating system formats all floppy disks with the FAT file system, because the additional files required for the NTFS file system use too much space on a floppy disk.
- If you format a removable disk with the FAT file system, you can remove the disk while Windows NT is running.
- When you format any removable disk with the NTFS file system, or if the page file is located on removable media, you can only remove the disk after you shutdown Windows NT.

Windows NT File Systems

This section contains information about the two file systems that Windows NT supports:

- File Allocation Table (FAT). The computer can access files on a FAT volume when it is running Windows NT, Windows 95, MS-DOS, or OS/2.
- Windows NT File System (NTFS). The computer must be running Windows NT to be able to access files on an NTFS volume.

The FAT file system is a simple file system originally designed for small disks and simple folder structures. The FAT file system is named for its method of organization, the file allocation table, which resides at the beginning of the volume. To protect the volume, two copies of the table are kept, in case one becomes damaged. In addition, the file allocation tables and the root folder must be stored in a fixed location so that the files needed to start the system can be correctly located.

The FAT file system works the same in Windows NT as it does in MS-DOS, Windows 3.1x, and Windows 95. In fact, you can install Windows NT on your existing FAT primary partition. When running Windows NT, you can move or copy files between FAT and NTFS volumes.

Note You cannot use Windows NT with any compression or partitioning software that requires disk device drivers to be loaded by MS-DOS. Therefore, you cannot use MS-DOS 6.0 DoubleSpace® or MS-DOS 6.22 DiskSpace on a FAT primary partition or logical drive that you want to access when running Windows NT.

The Windows NT file system (NTFS) provides a combination of performance, reliability, and compatibility not found in the FAT file system. It is designed to quickly perform standard file operations such as read, write, and search—and even advanced operations such as file-system recovery—on very large hard disks.

The NTFS file system includes security features required for file servers and high-end personal computers in a corporate environment. The NTFS file system supports data access control and ownership privileges that are important for the integrity of critical data. While folders shared on a Windows NT computer are assigned particular permissions, NTFS files and folders can have permissions assigned whether they are shared or not. NTFS is the only file system on Windows NT that allows you to assign permissions to individual files. However, when you move or copy a file from an NTFS to a FAT volume, permissions and other attributes unique to the NTFS file system are lost.

You can use either or both of these file systems on a computer. Which one you choose for your computer, or for an individual volume on your computer, depends on such things as:

- How you expect to use the computer.
- The hardware platform.
- The size and number of hard disks.
- Security considerations.

For more information about file systems, especially NTFS, see *Inside the Windows NT File System*, by Helen Custer (Microsoft Press 1994, ISBN 1-55615-660-X), which documents the NTFS file system design.

Maximum Sizes

The maximum file size and volume size for the FAT file system is 2^{32} bytes (4 GBs). FAT can support a maximum of 65,535 clusters per volume.

The NTFS architecture is designed to use numbers up to 2^{64} bytes (16 exabytes, or 18,446,744,073,709,551,616 bytes). In theory, it is possible to have a volume 2^{64} bytes in size. Even if there were hardware available to supply a logical volume of that capacity, there are other limitations to the maximum size of a volume.

Partition Tables are limited by industry standards to 2^{32} sectors. (For more information about Partition Tables, see the earlier sections titled "Partition Table" and "Using Hard Disks With More Than 1024 Cylinders (x86-based Computers)."

The sector size is a function of hardware and industry standards, and is normally 512 bytes. While sector sizes might increase in the future, the current size puts a limit on a single volume of 2 terabytes (2^{32} * 512 bytes, or 2^{41} bytes).

Windows NT provides the capability to combine discontiguous disk areas when creating volume sets, stripe sets, and stripe sets with parity, but these volumes have the same limitations. Even if there were physical disks with 2 terabyte (TB) capacity, they could not be combined to create larger volumes without increasing the physical sector size.

There is one case where you can combine disks that exceed the 2 TB limit. In Disk Administrator, there is an option to extend a volume set. You do this by selecting an existing primary partition, logical drive, or volume set that is already formatted as NTFS. You can then combine it with one or more areas of free space and select **Extend Volume Set** from the **Partition** menu. You end up with a larger NTFS volume set. However, if anything goes wrong with a volume set larger than 2 TB, you cannot just reformat it. You must again start with a smaller volume set (<=2 TB) and use the **Extend Volume Set** selection to extend the volume set to larger that 2 TB.

For the time being, 2 TB should be considered the practical limit for both physical and logical volumes using NTFS.

With the NTFS file system, you can have a maximum of 2^{32} clusters per file. This is an implementation limitation, not a file size restriction.

There is no specific limit to the number of files on an NTFS volume. However, you will not be able to create any new files when your volume is so full that Windows NT cannot allocate another entry in the MFT.

Which File System to Use on Which Volumes

This is a general recommendation for determining which file systems to use for which volumes:

- Use a small FAT primary partition (between 250 and 500 MB, depending on the total disk space and page file size) as the C drive. Use this partition for both your system and boot partition.
- Format the rest of your disk space as NTFS, and use the NTFS volumes for application programs and data. You might want to create more than one NTFS volume.

This arrangement gives you the following benefits:

- You have the recoverability of the NTFS file system for the most important information, your data. (You should always maintain your system and boot partitions by using backup tapes and creating an Emergency Repair Disk, as described in Chapter 5, "Preparing for and Performing Recovery.")
- FAT is more efficient for smaller volumes and NTFS is more efficient for larger volumes.
- NTFS provides file and folder security for your application programs and data:
- If you have a startup failure on an x86-based computer, this means that you can start up the computer with an MS-DOS bootable floppy disk to troubleshoot and recover from the problem.

These additional guidelines might affect your decision as to which file system(s) to use on your computers:

- For an x86-based computer, when Windows NT starts up, it looks for certain files in the root folder of the hard disk that contains the system partition. This partition can be formatted with either the NTFS or FAT file system. This partition should be large enough to accommodate all the files you need to access under that file system.
- For a RISC-based computer, the system partition must be formatted with the FAT file system. You can use the NTFS file system on your boot partition, which needs to be large enough for all Windows NT system components. If you configure your partitions in this way, your system partition should be 5-10 MB, and your boot partition should be at least 160 MB, 250-500 MB to allow for growth.
- Because you must format your system partition with the FAT file system on RISC-based computers, you can use Disk Administrator to secure the system partition. This prevents anyone who does not have administrative privileges from accessing the system partition in spite of the fact that it is formatted as FAT. See online Help for the Disk Administrator for more information.

Note The system partition contains the Partition Boot Sector and other files needed to load the operating system, such as NTLDR (for x86-based computers) and OSLOADER (for RISC-based computers). The boot partition needs to include the folder with the operating system. The boot partition and the system partition can be a single partition with other folders in it.

Controlling Access to Files and Folders

On NTFS volumes, you can set *permissions* on files and folders that specify which groups and users have access to them, and what level of access is permitted. NTFS file and folder permissions apply both to users working at the computer where the file is stored and to users accessing the file over the network when the file is in a shared folder. With NTFS you can also set *share permissions*, which operate on shared folders in combination with file and folder permissions.

Note To preserve permissions when you copy or move files between NTFS folders, use the Scopy program on the *Windows NT Server Resource Kit* CD.

With FAT volumes, you cannot set any permissions on the individual files and folders. The only security is share permissions that are set on the entire share, affect all files and folders on that share, and only function over the network. Once a folder is shared, you can protect the shared folder by specifying one set of share permissions that applies to users for all files and subfolders of the shared folder. Share permissions are set in very much the same way as file and folder permissions are set in NTFS. But because share permissions apply globally to all files and folders in the share, they are significantly less versatile than the file and folder permissions used for NTFS volumes.

Share permissions apply equally to NTFS and FAT volumes. They are enforced by Windows NT, not the individual file system.

Chapter 4, “Managing Shared Resources and Resource Security,” in the *Windows NT Server Concepts and Planning* book, contains several sections that describe setting and using file and folder permissions on FAT and NTFS volumes.

NTFS Compression

Windows NT supports compression on an individual file basis for NTFS volumes. Files that are compressed on an NTFS volume can be read and written by any Windows-based application without first being decompressed by another program. Uncompression happens automatically during the read of the file. The file is compressed again when it is closed or explicitly saved.

Compressed files and folders have an attribute of C when viewed in My Computer or Windows NT Explorer. You can select an alternate color for compressed files and folders by:

- On the **View** menu, click **Options**.
- On the **View** tab, check **Display compressed files and folders with alternate color**.

Only the NTFS file system can read the compressed form of the data. When an application like Microsoft Word for Windows or an operating system command like **Copy** requests access to the file, the NTFS file system uncompresses the file before making it available. For example, if you copy a compressed file from a another Windows NT computer to a compressed folder on your hard disk, the file will be uncompressed, copied, and recompressed.

This compression algorithm is similar to that provided by the MS-DOS 6.0 DoubleSpace® and MS-DOS 6.22 DriveSpace™ compression, with one important difference — the MS-DOS functionality compresses the entire volume, while the NTFS file system enables the user to compress individual files and folders in the NTFS volume.

NTFS compression is not supported for cluster sizes larger than 4K, because

- Compression causes a performance degradation.
- The reason you use very large (>4K) clusters is to improve performance in some special cases.

Therefore, you should not use compression and large clusters at the same time. In other words, compression performs reasonably well on sizes up to 4K, but beyond that size, the savings in disk space is not worth the decrease in performance. When the cluster size is >4K on an NTFS volume, none of the NTFS compression functions are available.

Other compression utilities are available to compress files on computers running Windows NT. These utilities differ from NTFS compression in the following ways:

- They usually can be run only from the command line.
- Files cannot be opened when they are in a compressed state — the file must first be uncompressed by using the companion program to the one used to compress the file. When you close the file, it is saved in an uncompressed state, and you have to use a program to compress it.

The *Windows NT Server Resource Kit* includes a compress program and an expand program. These compress program can only be run from the command line. The expand program is a Windows-based application. There is also an expand program on the Windows NT Server product CD, which you need to run from the command line. For more information about these utilities, see "File System Utilities" in Chapter 10, "Disk, File System, and Backup Utilities."

As described earlier, the DoubleSpace and DriveSpace compression features in MS-DOS cannot be used when running Windows NT.

NTFS Recoverability

A lazy-write file system, such as the FAT file system when running Windows NT, uses an intelligent cache-management strategy and provides a way to recover data (such as the Chkdsk program) if there is an error when writing to the disk. All data are accessed via the file cache. While the user searches folders or reads files, data to be written to disk accumulates in the file cache. If the same data are modified several times, all those modifications are captured in the file cache. The result is that the file system needs to write to disk only once to update the data.

NTFS is *recoverable file system.* It combines the speed of a lazy-write file system with virtually instant recovery.

The NTFS file system guarantees the consistency of the volume by using standard transaction logging and recovery techniques, although it doesn't guarantee the protection of user data. It includes a lazy-write technique plus a volume-recovery technique that takes typically only a second or two to insure the integrity of all NTFS volumes each time the computer is restarted.

Effects of File and Disk Cache on Recoverability

The file cache is just an area of RAM that contains the data. When you write data to disk, Windows NT's lazy-write technique says that the data are written, when, in fact, they are still in RAM.

There can also be cache memory on the disk controller or on the disk itself. If you are running MS-DOS or Windows 3.1x, using the disk or controller cache can really help performance. Windows NT's use of the lazy-write technique improves performance to the point where you might not see much more improvement by using the cache on the disk or the controller.

This information should help you decide whether you want to enable the disk or controller cache:

- If disk performance is an issue, and specifically if the disk is being heavily written, turning on write caching is an option for improving performance, especially if it can be measured to improve the performance of the system.
- Controlling the write-back cache is a firmware function provided by the disk manufacturer. See the documentation supplied with the disk. You cannot configure the write cache from Windows NT.
- Write caching does not impact the reliability of the file system's own structures (metadata). NTFS instructs the disk device driver to insure that metadata writes get written through regardless of whether write caching is enabled or not. Non-metadata is written to the disk normally, so caching can occur for such data.
- Read caching in the disk has no impact on file system reliability.

Automatic Recovery from Disk Errors

This section discusses how Windows NT automatically recovers from some kinds of disk errors. You do not even know that an error occurred.

Windows NT provides two kinds of disk error recovery:

- Dynamic data recovery by using sector sparing, which is only available on SCSI disks that are configured as part of a fault-tolerant volume. Sector sparing works on fault-tolerant volumes because a copy of the data on the sector with the error can be regenerated.
- NTFS cluster remapping.

Windows NT Server provides additional recovery mechanisms for fault-tolerant volumes (mirror sets and stripe sets with parity). For more information about fault-tolerant volumes, see Chapter 5, "Preparing for and Performing Recovery."

The following table summarizes what happens if a sector goes bad on a computer running Windows NT Server. There is more information about sector sparing and cluster remapping following the table. For more details, see Chapter 5, "Volume Management and Fault Tolerance" in the book *Inside the Windows NT File System.*

Description	FtDisk installed with a SCSI disk that has spare sectors	FtDisk installed with a non-SCSI disk or disk with no spare sectors	FtDisk not installed with any kind of disk
Fault-tolerant volume (Windows NT Server only)	1. FtDisk recovers the data. 2. FtDisk replaces the bad sector. 3. File system doesn't know about the error.	1. FtDisk recovers the data. 2. FtDisk sends the data and bad-sector error to the file system. 3. NTFS performs cluster remapping. 4. FAT doesn't do anything about the error.	N/A
Non-fault-tolerant volume	1. FtDisk can't recover the data. 2. FtDisk sends a bad-sector error to the file system. 3. NTFS performs cluster remapping. On a read operation, data are lost. 4. FAT loses the data on both read and write.	1. FtDisk can't recover the data. 2. FtDisk sends a bad-sector error to the file system. 3. NTFS performs cluster remapping. On a read operation, data are lost. 4. FAT loses the data on both read and write.	1. Disk driver returns a bad-sector error to the file system. 2. NTFS performs cluster remapping. On a read operation, data are lost. 3. FAT loses the data on both read and write.

Note FtDisk is the device driver that handles I/O to volume sets, stripe sets, mirror sets, and stripe sets with parity. However, volume sets and stripe sets are not fault tolerant, so processing of an error on one of these volumes falls into the non-fault-tolerant volume category.

Sector Sparing

If a disk read error occurs on a file on a fault-tolerant volume on a hard disk connected to a SCSI controller, Windows NT Server uses a technique called *sector sparing* to recover the data. Windows NT Server obtains a spare sector from the disk device driver to replace the bad sector. It then recovers the data by reading the sector from the mirror disk or recalculating the data from a stripe set with parity, and writes the data to the new sector.

This processing is transparent to any applications performing disk I/O. Sector sparing eliminates error messages such as the “Abort, Retry, or Fail?” that occur when a file system encounters a bad sector.

Cluster Remapping

With transaction logging and recovery, NTFS guarantees that the volume structure will not be corrupted, so all files remain accessible after a system crash. However, user data can be lost because of a system crash or a bad sector.

The NTFS file system implements a recovery technique called cluster remapping. When Windows NT returns a bad sector error to the NTFS file system, NTFS dynamically replaces the cluster containing the bad sector and allocates a new cluster for the data. If the error occurred during a read on a non-fault-tolerant volume, NTFS returns a read error to the calling program, and the data are lost. When the error occurs during a write, NTFS writes the data to the new cluster, and no data are lost.

NTFS puts the address of the cluster containing the bad sector in its Bad Cluster File so the bad sector will not be reused.

Using MS-DOS-based Disk Utilities

Do not use any third-party MS-DOS-based disk program to repair or defragment a FAT volume used by Windows NT unless the disk program has been certified as being compatible with Windows NT version 3.5 or later. While instances of third-party utilities corrupting FAT volumes are relatively rare, some problems have occurred. Using non-Windows NT utilities are simply not worth the risk of losing an entire volume. Even the MS-DOS-based program Defrag might not work correctly if you have long filenames in the FAT volume.

You can use the Windows NT command prompt program Chkdsk to scan and repair FAT and NTFS volumes. This program incorporates all of the functionality of the MS-DOS-based Chkdsk and Scandisk utilities, including a surface scan. To do a surface scan, use **chkdsk /r**.

There is a Windows NT-based program available that can defragment both FAT and NTFS volumes. Microsoft has a directory that contains information about hardware and software programs that are available for Windows NT, including defragmentation utilities. This directory is called InfoSource. For information about InfoSource, see http://www.microsoft.com/infosource.

Disk media verification can be done with the SCSI utilities that ship with most SCSI controllers. Consult the documentation accompanying your hardware or contact your hardware vendor for information on verification of disk media. It is always a good idea to do a complete backup of the disk before starting a procedure of this type.

Generating and Viewing Short Filenames

Filenames on Windows NT platforms can be up to 256 characters, and can contain spaces, multiple periods, and special characters that are illegal in MS-DOS filenames. These *long* filenames use the 16-bit Unicode character set.

Windows NT makes it possible to access files with long names from other operating systems by automatically generating an MS-DOS–readable (eight-plus-three) name for each file. This way, files are accessible over a network by computers using the MS-DOS, Windows 3.1x, and OS/2 operating systems, as well as by computers using Windows NT and Windows 95 operating systems. When an application saves a file on a computer running Windows NT, both the eight-plus-three filename and long filename are retained.

Note Use caution with MS-DOS-based or Windows 3.x-based applications when running under Windows NT. With these applications, if you save a file to a temporary file, delete the original file, and rename the temporary file to the original filename, the long filename is lost. Any unique permissions set on that file are also lost.

If the long name of a file or folder contains spaces, be sure to surround the name with quotation marks. For example, if you have a program called DUMP DISK FILES that you want to run from the Start icon and you enter the name without quotation marks, you will get an error message that says "cannot find the program DUMP or one of its components."

You must also use quotation marks when a path typed at the command line includes spaces, as in the following example:

```
move "c:\This month's reports\*.*" "c:\Last month's reports"
```

Use wildcards such as * and ? carefully in conjunction with the **del** and **copy** command prompt commands. Windows NT searches both long and short filenames for matches to the wildcard combination you specify, which can cause extra files to be deleted or copied.

To copy or move files with case-sensitive long filenames, it is safest to select the files using a mouse in Windows NT Explorer and My Computer. That way, you can clearly identify which files you want to copy or move.

Because both the FAT file system and the NTFS file system use the Unicode character set for their names, there are several illegal characters that MS-DOS cannot read in any filename. To generate a short MS-DOS–readable filename for a file, Windows NT deletes all of these characters from the long filename and removes any spaces. Since an MS-DOS-readable filename can have only one period, Windows NT also removes all extra periods from the filename. Next, Windows NT truncates the filename, if necessary, to six characters and appends a tilde (~) and a number. For example, each nonduplicate filename is appended with **~1**. Duplicate filenames end with **~2**, **~3**, and so on. Filename extensions are truncated to three or fewer characters. Finally, when displaying filenames at the command line, Windows NT translates all characters in the filename and extension to uppercase.

When there are five or more files that would result in duplicate short filenames, Windows NT uses a slightly different method for creating short filenames. For the fifth and subsequent files, Windows NT:

- Uses only the first two letters of the long filename.
- Generates the next four letters of the short filename by mathematically manipulating the remaining letters of the long filename.
- Appends ~1 (or another number, if necessary, to avoid a duplicate filename) to the result.

This method provides substantially improved performance when Windows NT must create short filenames for a large number of files with similar long filenames. Windows NT uses this method to create short filenames for both FAT and NTFS volumes.

For example, these are the long and short filenames for six files that you create in the order test 1 through test 6.

Long filename	**Short filename**
This is test 1.txt	THISIS~1.TXT
This is test 2.txt	THISIS~2.TXT
This is test 3.txt	THISIS~3.TXT
This is test 4.txt	THISIS~4.TXT
This is test 5.txt	TH0FF9~1.TXT
This is test 6.txt	THFEF5~1.TXT

However, when you create the files in the order shown in this table, you get the following short filenames.

Long filename	**Short filename**
This is test 2.txt	THISIS~1.txt
This is test 3.txt	THISIS~2.txt
This is test 1.txt	THISIS~3.txt
This is test 4.txt	THISIS~4.txt
This is test 5.txt	TH0FF9~1.txt
This is test 6.txt	THFEF5~1.txt

Windows NT displays the long names for folders and files. You can use Windows NT Explorer and My Computer to see the short name by selecting the file or folder and selecting **Properties** on the **File** menu.

From the command line, to see both the long and short filenames for each file in the folder, type the following command:

```
dir /x
```

Tip To display both long and short filenames automatically when using the **dir** command, use the System option in Control Panel to set the **dircmd** variable to the value **/x**.

CHAPTER 4

Planning a Reliable Configuration

This chapter discusses planning what to do when failures occur from the viewpoint of:

- Deciding what hardware and software to procure.
- Developing your plan for doing backups.
- Developing training plans, disaster plans, and contingency plans.

Chapter 5, "Preparing for and Performing Recovery," discusses what information about your computers and which data on your computers you should back up. It also describes using these backups to recover from failures.

When planning an installation that includes computers running Windows NT Server and Windows NT Workstation, consider questions such as the following:

- How many computers running Windows NT Server and how many computers running Windows NT Workstation do you need for your enterprise?
- Are there computers running operating systems other than Windows NT Server and Windows NT Workstation in your enterprise?
- Where should the computers be located, and how should they be connected?
- How many single points of failure can you tolerate in your enterprise?
- What fault tolerance methods should you implement?
- Which data on the computers needs to be backed up, and what is your backup strategy?
- What records do you need to maintain to be able to troubleshoot problems or recover from failures?
- Where are you going to store your backups and other records necessary to your operation?

For each individual computer that is going to be using Windows NT Server or Windows NT Workstation, consider questions such as the following:

- What file system(s) should you use?
- How many controllers and disks do you need, and how should they be configured?
- Should some of your data be configured on fault-tolerant volumes?

  If so, should you buy fault-tolerant hardware that can be used on a computer running Windows NT Workstation or Windows NT Server?

  Or do you want to use the fault-tolerant software features available in Windows NT Server?

Another important topic to consider is disaster planning. Have you developed and tested solutions to occurrences such as fires, floods, and power outages before a disaster occurs?

Planning Considerations

This section discusses topics that you should consider in your planning and describes where you can find more information.

Planning Approaches

There are three ways that you can approach planning:

- Top-down execution and responsibility.
- Bottom-up execution and responsibility.
- Top-down policy and bottom-up planning and execution.

In the top-down execution and responsibility, one department in the company is responsible for planning and coordination for all departments. With this method, you might have a centralized operations staff that is responsible for all or most of the computers running Windows NT Server, and for doing all of the backups. This method enables you to create a group with specialized, expert knowledge for developing procedures, procuring and installing hardware, doing testing, and training personnel. The disadvantage to this approach is that the specialized group might not understand how a particular department's data relates to the overall business. Each department is just one of many, and therefore might not get the appropriate attention.

In the bottom-up execution and responsibility, each department is responsible for developing its own plans and implementing them. Each department would probably be responsible for all of the computers that they use, and would develop their own backup procedures. The impact of the loss of a department's data is better understood by the department's management. However, this approach might cost more, because each department needs to have its own personnel and to do its own research.

In top-down policy and bottom-up planning and execution, one department or group develops the overall planning guidelines. Each department is responsible for developing its own plans and procedures. The individual departments are also responsible for implementing their procedures. With this method, there is a consistent approach, and each department knows what details are important to its successful operation.

Whichever approach you use, continuous improvement and updates are better than total rewrites as new information, policies, procedures, and the like become available.

Security

An important area to consider in your planning is minimizing the effects of human error or deliberate sabotage attempts. If anyone can walk up to your computer running Windows NT Server and restart it, no amount of security that you implement by using software can protect your computer. Not only can they damage information on the computer, but they can steal information from your computer.

You can implement procedures that restrict peoples' physical access to your facility, or to only those areas to which they should need access. Put your computers in a secure room. Lock them. Use a password on your screen saver. Run virus checks on floppy disks before you use them. Or disable the floppy disk, which you can sometimes do by using BIOS options. Otherwise, you can physically disconnect it. Run virus checks on your computer.

Windows NT enables you to manage what your users can and cannot do by creating profiles for each of your users and restricting their access to files and servers. But no amount of planning will cover all of the ways people can cause damage to data on your computers or the computers themselves. Consider security when you develop your backup and storage procedures.

Windows NT Server includes software to write information to the security event log and to audit the log for possible attempts to breach security. You can log things like:

- Access to files.
- Invalid logon attempts.
- All logons.

The Audit policy controls what types of events are recorded in the security log.

Chapter 2, "Network Security and Domain Planning," in the *Windows NT Server Networking Guide*, contains information about security policies. Chapter 3, "Managing User Work Environments," and Chapter 4, "Managing Shared Resources and Resource Security," in the *Windows NT Server Concepts and Planning* book, contain more information about security features. Chapter 6, "Windows NT Workstation Security Model," in the *Windows NT Workstation Resource Guide*, contains more information about physical security as well as security features available when you are running Windows NT. There is also more information in Windows NT Help.

Hardware and Software Upgrades

Expect to do hardware and software upgrades. Planning when and how to do the upgrades can minimize the time they take and the risk that something will go wrong.

This sections lists things to consider in planning for upgrades. There is more information about many of these topics in other sections of this chapter. Chapter 5, "Preparing for and Performing Recovery," and Chapter 6, "Troubleshooting Startup and Disk Problems," also contain more information about many of these topics.

When doing hardware or software upgrades:

- Never upgrade without backing up data files that you must have.
- Be sure to back up the Registry, Master Boot Record, Partition Boot Sector, and other critical system data.
- Have a current Emergency Repair Disk for each computer.
- Have records of the configuration of the computer, such as jumper and DIP switch settings, IRQs, and the hard disk configuration.

The *Windows NT Server Resource Guide* and the *Windows NT Workstation Resource Guide* contain information about deploying Windows NT.

Maintenance and Record Keeping

Routine, planned maintenance can prevent problems or minimize their effects. For example, if there are file system problems, you might not know about them until you restart the computer, when the Chkdsk program tells you that there are errors. Backups can even be corrupt in some situations, because no backup program, including Windows NT Backup, can detect corrupt user data.

Administrators should check both the system and application event logs on a daily basis. Impending file system problems might show up as errors in the system log before the file system is corrupt. Be sure to investigate any messages in the event log from FtDisk or disk device drivers.

Disk fragmentation can cause performance problems. You should consider running a defragmentation program on a regular basis. There are third-party defragmentation utilities that run under Windows NT and can defragment both FAT and NTFS volumes. Be sure to only use utilities that are designed for Windows NT. Microsoft has a directory that contains information about hardware and software programs that are available for Windows NT, including defragmentation utilities. This directory is called InfoSource. For information about InfoSource, see http://www.microsoft.com/infosource.

Create a log book for every computer, which should contain information about the computer's configuration. To make recovery easier, there are several system files that you should back up every time you make certain changes to your Windows NT configuration. For information about these topics, see the section "Maintaining Configuration and Essential System Information" in Chapter 5, "Preparing for and Performing Recovery."

Maintaining Reference Information

Create a technical reference library for all hardware and software documentation. This library can also include other documentation related to your computers that you want to have in a central location. The types of information for this library are:

- Vendor documentation, including manuals, receipts, warranties, proofs of purchase, and the like.
- Insurance policy and any claims.
- Information about any kits, tools, and add-ons that you have installed.
- All internal documentation that you have generated, including policies, procedures, training guides, and the like.
- Hardware configuration information for each of the computers, including, but not limited to, the following:
    - Their location, cabling, intermediary devices, and external connections.
    - Computer type, model number, serial number, motherboard, and CPU.
    - Computer BIOS manufacturer and revision level (x86-based computers) or firmware revision level (RISC-based computers).
    - CMOS information for x86-based computers.
    - Firmware information on RISC-based computers.
    - For each component, information such as IRQ, DMA addresses, and I/O port.
    - Amount of RAM, parity or non-parity.
    - Information about the video display and the video card.
    - Complete configuration of the disk subsystem, including the makes, models, and serial numbers of all disks and controllers, and their type (such as SCSI or EIDE).
    - Configuration information for other storage media, such as tape, optical disk, or removable disk.
    - Types and configuration of network cards.
    - Configuration information about any other adapters, such as multiport adapters.
- Software configuration information and backups for each computer, including:
    - Disk map, describing location and size of each partition and logical drive.
    - Backups of each Master Boot Record and Partition Boot Sector.
    - Windows NT startup floppy disk.
    - Windows NT versions installed and the partitions on which they are installed.

  - User information, including which users are allocated disk space and what utilities and application programs they are using.
  - Backups of the system configuration files, including the Registry.
  - Which application programs are installed on which volumes.
  - Licensing information.
- Phone List
  - Notification plan of who to contact for which kinds of problems or emergencies, and how to notify them.
  - Administrators.
  - Vendor and consultant support.
  - Management.
  - Critical Users.

Disaster Planning

Just as every business needs to plan and budget for future growth, you need to have a plan for dealing with total or partial loss of business data.

In order to determine what provisions to make for partial or complete loss of data, you need to determine the approximate dollar and manpower cost of rebuilding or replacing the data your company uses. You should consider the following questions:

- Do you know the cost of reconstructing your company's financial, personnel, and other business data?
- Do you know if your business insurance would cover any or all of the cost of replacing data?
- Do you know how long it would take to reconstruct your business data? How would this translate into lost future business?
- Do you know the cost per hour of server downtime?

To prevent a natural disaster or sabotage from becoming an even worse financial disaster for your business, test your plan for recovering and restoring your company's critical data. You should keep copies of your disaster recovery plan both on site and off site, and key personnel should consider keeping a copy at home.

Because there are many articles in magazines and entire books that discuss disaster planning and recovery, this section does not go into detail about any aspect of disaster planning. Instead, it describes topics that you should think about in your planning. Your insurance company should also be able to provide you with information to help you.

There are several areas that need to be addressed in developing a comprehensive disaster recovery plan. Some of the topics that you should consider in your disaster planning are also topics that you should follow in your day-to-day operations.

A comprehensive disaster plan should address all of the following issues:

- What data do you need to back up and how often should you do backups?
- Saving critical computer or other hardware configuration information that is not saved during normal tape backups.
- What data needs to stored on site, and how should it be physically stored?
- What data should be stored off site, and how should it be physically stored?
- What training is required so server operators and administrators can respond quickly and effectively if an emergency occurs.

There is more information about each of the preceding topics in this chapter and in Chapter 5, "Preparing for and Performing Recovery."

Contingency Planning

This section identifies topics you should consider in your contingency planning.

Identifying Costs of a Failure

There are several ways to measure costs. Some costs are easy to understand and to calculate, such as the following:

- Replacement costs for file servers, mail servers, or print servers.
- Replacement costs for servers running applications such as Microsoft SQL Server or Systems Management Server (SMS).
- Replacement costs for gateway servers running Microsoft RAS, SNA, Proxy, or NetWare.
- Workstation replacement costs for personnel in different departments.
- Replacement costs for individual computer components, such as a hard disk or a network card.

But measuring the cost of server downtime to a department or your company can be more difficult.

And how do you measure the cost of a server failure on:

- Lost sales.
- Lost customer goodwill.
- Lost employee productivity and confidence.
- Increased costs because of makeup time.
- Missed contractual obligations or possible legal liabilities.
- Perishable products going to waste.
- Loss of competitiveness.

If you have kept records of failures, you might find them useful in your contingency planning. You can investigate ways to avoid each failure, or to minimize the downtime associated with the failure. If you have cost information for the failures, you can then compare the cost of each failure to the cost of preventing or minimizing the failure.

Here are two examples:

Failure	File server in sales department down, network card failure.	Router failure between development and testing department.
Effect	Lost sales	Lost productivity of employees.
Total downtime last year.	3 hours.	16 hours.
Costs of failure per hour.	$10,000	Average hourly wage of 10 affected employees is $18/hr.
Total downtime costs last year.	$30,000	$2,880
Possible resolution or workaround.	3 spare network cards @ $500 each.	Put an alternate router in place or obtain a spare router.
Expected costs of resolution or workaround.	$1,500	$500 - $2,000
Estimated savings during first year with resolution in place.	$28,500	$880-$2,380

What Components in the Computer Might Fail

A server failure is typically the most costly with respect to a corporation's business, whether it is a file server, a print server, or an applications server. This section discusses the components within the computer to help you decide how to configure it. You should also regularly run diagnostics on the individual components.

Motherboard and CPU (Central Processing Unit)

Motherboards consist of electronics that can and do fail, yet the motherboard and the CPU are the more reliable computer components. There is not a lot you can do to avoid a motherboard failure or CPU fault, except to regularly run system checks that ensure they are functioning correctly. Some vendors provide systems having built-in diagnostics that operate with Windows NT.

RAM

There are three major types of RAM, in the sense of error detection and correction.

- Parity RAM. Parity RAM has an extra bit that indicates if each byte in the RAM is good or faulty. When parity RAM detects a parity difference, it signals the CPU through a Non-Maskable Interrupt (NMI). Depending on where and when this happens, Windows NT determines if this is an I/O board parity error, memory bus error, or some other kind of parity error.
  Windows NT can also report I/O channel parity errors from cards in slots. You get an error message in these cases, and sometimes the computer stops.
- Error Corrective Coding (ECC) RAM. High-end systems often use ECC RAM, which can detect a two-bit failure and correct a single-bit failure in the system memory. Windows NT continues to run in spite of a single-bit failure. Depending on the hardware vendor's design, there might or might not be a report of this corrective action.
- Non-parity RAM. If you are using non-parity RAM, Windows NT has no way to detect memory problems, and your computer might crash randomly and inconsistently. Non-parity RAM is cheaper, and parity RAM is not available for all computers. If you do not have parity RAM in your computers, ask your vendors if parity RAM can be installed or supported by the computer.

There are a few vendors that supply products that you can use to check the RAM in a computer.

Video Cards

Video cards drive the screen as well as render images for display. Video cards rarely will cause a computer to crash, but rather, might cause the computer to behave erratically, which can be confusing to diagnose. More often, video cards cause screen redraw problems, application page faults, and the like. These problems are usually not critical enough to require you to shutdown the computer. To minimize video problems, be sure that your computer is running with the most recent release of a supported video driver from Microsoft, or the third-party vendor selling that card.

Disks and Disk Controllers

You should investigate IDE, EIDE, and SCSI technologies, because each offers different benefits with respect to fault tolerance and recovery. You have many choices for your disk configuration, including fault-tolerant configurations. The meantime between failures (MTBF) gives you a measure of expected reliability of disks and controllers.

Be sure to run disk and controller diagnostics during every preventive maintenance period. Diagnostics should be available from your hardware vendor. Windows NT automatically runs its Chkdsk program every time you start up, and you can run a surface scan of the disks by specifying **chkdsk /r**.

Backup Devices

Verifying your backups by doing test restores is the best way to make sure that your backup devices and media are working correctly.

Network Cards

FDDI, CDDI, and ATM network cards can have dual-channel connections. If one channel goes down, the other channel is automatically used.

Ethernet and token ring network cards do not have a dual-channel capability. If the manufacturer provides a diagnostic program, you should run diagnostics on the network cards during scheduled preventive maintenance or down-time periods.

You can evaluate network segments with network packet trace programs, called sniffers. The Windows NT Network Monitor can check for:

- Bad cyclic redundancy checks (CRC).
- Corrupted packets.
- Bandwidth saturation caused by a broadcast intensive network card.

Power Considerations

You need to have consistent, reliable power to be able to run your computers. Power failures, power surges, and power sags can cause the computers to crash and can damage the electronics. There are different situations that affect the power supplied to your computer.

Power Supply

Be sure your computer has a high quality power supply that can simultaneously support all components attached to it. It is possible to overload a power supply by adding too many power consuming devices to a computer. It is also possible to overload a circuit by having too many computer components on it.

Power Outages

To protect against damage and loss of data from temporary power outages, consider the purchase of an uninterruptable power supply (UPS). Windows NT supports different kinds of UPSs and can send messages to users to save their data and to log off as soon as the UPS device signals an impending shutdown. For more information about the Windows NT UPS service, see the section titled "Avoiding Single Points of Failure," presented later in this chapter.

Power Surges

To prevent power surges that can destroy your data, obtain quality power surge surpressors.

What Components in the Network Might Fail

If users cannot connect to your computers running Windows NT Server, you do not have a fault-tolerant configuration. You need to consider what can fail in the connections between your computers as well as the individual computers themselves.

Network Cabling

If your company is growing, the connections between your computers could become saturated with network traffic. You should evaluate network traffic regularly to determine if you will need to upgrade with more equipment. You can also use newer technologies like FDDI, CDDI, ATM, Fast Ethernet 100 Base T, and the like.

Intermediary Devices

Devices that connect different segments of your network, such as routers, bridges, hubs, and switches, can also be bottlenecks and points of failure. You should have UPS protection for these devices.

For each of these devices, find out whether you can get the vendor support you need. What standards are supported by each of the devices, and does the vendor have a migration path for new standards? You should also find out if there are frequent software changes.

External Network Connections

If you lease an X.25, ISDN, or T1 line to connect to another building, branch, or subsidiary of your company, verify that your line vendor has recovery procedures in place that can guarantee minimum down times for the line.

Electric Wiring

Be sure the wiring in your building is capable of supplying your company with enough power as the demands in electricity increase. This is especially important if the building is older or if there are other companies in the building that are putting more demands on the supply of electricity.

What Else Might Fail

Climate Control

If the weather at the location of your company requires heating or cooling within the building to keep your computers and network devices within required operating temperatures, consider making the climate control system fault tolerant as well.

Software Failures

Does your software vendor provide the support you need in case of software failure? Does your company have a technical support group to assist users when there are software problems?

How Likely is a Failure

The meantime between failures (MTBF) information supplied by some manufacturers of equipment such as hard disks is unlikely to provide useful information without extensive analysis and modeling according to the variables that exist in your company's usage pattern. You can use the MTBF as a relative measure of reliability.

More useful is records of past failures and their causes, because you can use this information to help you in your planning. You can categorize failures by their type, such as:

- Hardware failure of a server, client, or network component.
- Software failure of the operating system on the server or client, or an applications failure.
- Administrative error or oversight.
- User error.
- Deliberate damage, such as sabotage or viruses.

These are some questions to ask about failures:

- Have you taken any actions to reduce the likelihood of each failure occurring in your business?
- What was done or could be done to fix the problem?
- How long would it or did it take?
- What would it or did it cost?
- What changes have you made that might result in more or fewer failures?
  - Number of servers.
  - Number of clients.
  - Number of users.
  - Number of administrators.
  - Number of intermediary devices.
  - Number of external connections.
  - Physical size of LAN(s) or WAN.

Fewer computers can be easier to manage than many computers. However, the relative downtime impact is higher when you have more users connected to a smaller number of servers.

Importance of Training and Technical Support

Having trained personnel can reduce the likelihood of failures and reduce their severity. However, you need to determine if the cost of the training will be worth the expected benefit.

There are several ways you can train your personnel and provide them with technical support:

- Use self-study courses.
- Subscribe to TechNet.
- Use the Internet to access Microsoft and vendor information.
- Take vendor-approved or third-party courses.
- Have personnel become certified in the use, administration, and troubleshooting of system hardware and software.
- Have a technical library available for personnel.
- Install computers to be used specifically for training and testing.
- Develop your own training courses.

You can also contract to Microsoft, your hardware vendors, and third-party consultants for support.

See the *Windows NT Server Start Here* book for a description of Microsoft's AnswerPoint Information Services. Microsoft's support offerings range from no-cost and low-cost online information services (available 24 hours a day, 7 days a week) to annual support plans.

Importance of Testing

Testing is an important component of your contingency planning. You can use testing to try to predict failure situations and to practice recovery procedures. Be sure to do stress testing and test all functionality.

The following list identifies some of the failures that you should test:

- Individual computer components such as hard disks and controllers, processors, and RAM.
- External components such as routers, bridges, switches, cabling, and connectors.

These are some of the stress tests that you should set up:

- Heavy network loads.
- Heavy disk I/O to the same disk.
- Heavy use of file, print, and applications servers.
- Large number of simultaneous logons.

Avoiding Single Points of Failure

Minimize the number of components whose failure will result in the failure of the computer. The "Contingency Planning" section, presented earlier in this chapter, discusses components and reducing the likelihood of failure.

Computers running Windows NT Server have *fault-tolerance* features built-in to the operating system. Fault tolerance is the ability of a system to continue functioning when a component on the computer fails. Normally, the expression fault tolerance is used to describe disk subsystems, but it can also apply to other parts of the system or the entire system. Fully fault-tolerant computers use redundant disk controllers and uninterruptable power supplies (UPS) as well as fault-tolerant disk subsystems.

Although the data are always available and current in a fault-tolerant disk configuration, you still need to make backups to protect the information on your disk subsystem from:

- User errors.
- Sabotage.
- Software malfunctions.
- Catastrophic disasters such as fire, earthquakes, tornadoes, and floods.

Disk fault tolerance is not an alternative to a backup strategy with offsite storage. For more information about disk fault tolerance, see "Planning a Fault-tolerant Disk Configuration," presented later in this chapter.

Consider having replacement disks and controllers available on site. For instance, SCSI controllers cost as little as a few hundred dollars, while the cost of 50 to 500 users who cannot work while waiting for a replacement could be many thousands of dollars.

Consider providing UPS protection for individual computers and the network itself, including hubs, bridges, and routers. Windows NT has UPS support on individual computers. These UPSs typically provide power for five to 20 minutes, long enough for Windows NT to do an orderly shutdown when power fails. If there is a history of frequent or prolonged power outages, you should investigate ways to provide power for your critical computers other than from your local power company. It is important to remember that individual UPS systems, even for every computer system on the network, will not necessarily prevent data loss or corruption due to power fluctuations. The network is itself an electrical system. Intermediary devices such as routers, bridges, and hubs, require the same UPS protection in order to prevent loss of network functionality.

There is more information about UPS in Chapter 7, "Protecting Data" in the *Windows NT Server Concepts and Planning* book, and in Chapter 5, "Preparing For and Performing Recovery," in this book.

Backup Strategy

Doing regular backups of data on servers and local hard disks prevents data loss and damage caused by disk failures, power outages, virus infection, and network problems. Backup operations based on careful planning and reliable equipment make file recovery easier and less time consuming.

Windows NT includes the backup program, Ntbackup.exe, that enables you to use a tape drive to back up and restore files on Windows NT file system (NTFS) or file allocation table (FAT) volumes. The Ntbackup program also simplifies archiving. You can easily save data for legal or historical purposes and remove older, unused files, knowing that you can recover them, if necessary.

Ntbackup can perform several different kinds of backups:

- A *normal* backup, also called a *full* backup, copies all selected files and marks each as having been backed up. With normal backups, you can restore files quickly because files on the last tape are the most current.
- A *copy* backup copies all selected files but does not mark each file as having been backed up. Copying is useful if you want to back up files between normal and incremental backups, because copying does not invalidate these other backup operations.
- An *incremental* backup backs up only those files created or changed since the last normal or incremental backup. It marks files as having been backed up. If you use a combination of normal and incremental backups, restoring requires starting with your last normal backup and then working through all the incremental tapes.
- A *differential* backup copies those files created or changed since the last normal (or incremental) backup. It does not mark files as having been backed up. If you are doing normal and differential backups, restoring requires only the last normal and last differential backup tape.
- A *daily* backup copies all selected files that have been modified the day the daily backup is performed. The backed up files are not marked as having been backed up. (This can be useful if you want to take work home and need a quick way to select the files that you worked on that day.)

Chapter 6, "Backing Up and Restoring Network Files," in the *Windows NT Server Concepts and Planning* book, describes developing your backup plan and using Ntbackup.

When developing your backup plan:

- Be sure to have spare hardware in case of a failure in your backup device.
- Plan to test your backed up data regularly to verify that you can rely on your backup procedures and equipment.
- Include stress testing of your backup hardware (tape drives, optical drives, and controllers) and software (backup program and device drivers).
- Consider security regarding your backups. If you have implemented physical security to protect your computers, but unauthorized people can get access to your backup tapes, your data are not protected.

This section:

- Provides an approach for backing up computers on a small LAN and discusses backing up very large servers and ones that must be available 24 hours a day.
- Discusses the types of data that you should regularly back up.
- Lists questions that you should consider when planning your backup procedures.

Backup Approaches

There are two extremes when it comes to backing up data on computers running Windows NT:

- You have a small, simple network.
- The computer that you want to back up needs to be available 24 hours a day, 7 days a week, or contains a large amount of data to back up.

This section discusses backup approaches for these two situations.

Small and Medium LANs

This section describes an approach for backing up computers running Windows NT Server and Windows NT Workstation on a small to medium LAN. There are no backup devices connected to computers running Windows NT Workstation. Therefore, data from those computers needs to be backed up on computers running Windows NT Server.

This network has the following characteristics:

- 1 - 10 computers running Windows NT Server.
- Total disk space 1 - 25 GB on each computer running Windows NT Server.
- Typical daily data modifications in the range of 100-1000 MB.
- Typically has scheduled downtime at night or on weekends.
- There are a minimum of two backup devices on different computers running Windows NT Server. One should be on the primary domain controller (PDC). If you have backup domain controller (BDC), this computer should have the second backup device.
- All computers running Windows NT Server are capable of hosting the backup devices.

These are suggested guidelines for backing up all of the computers that are running Windows NT, assuming that you are using Ntbackup or a similar program to back up your data to a cartridge tape:

- If you are using tape drives that are SCSI controlled, insure that the drives will work in all of the computers. Are there spare SCSI IDs available on the chain?
- Does each computer still function correctly after you install the tape drive?
- Does Ntbackup work on each computer, and does a test restore work on each computer?
- All backup devices should use the same media. Preferably, all backup devices are the same make and model.
- Verify that the media works in all of the backup devices.
- The backup device must be fast enough to complete a backup in the time allotted, which could be overnight, or a specified number of hours.
- Perform a complete backup (called normal or full in NTBackup) on a regular basis, such as once a week.
- Do an incremental backup of files on the days that you do not perform a full backup.
- The backup media should be capable of writing a complete incremental backup on one cartridge. Preferably, a full backup fits on one cartridge.
- Label all backup media with the date, the computer on which it was created, and the type of backup, such as full or incremental.

- Rotate your backup media. Some suggestions are described in Chapter 6, "Backing Up and Restoring Network Files," in the *Windows NT Server Concepts and Planning* book.
- Secure the backup media in a fireproof safe large enough for a full rotation's worth of cartridges.
- Split nightly backups on all computers running Windows NT Server among the backup devices for load balancing and reliability.
- Back up the Registry on your PDC and BDC every night.
- Back up critical data located on computers running Windows NT Workstation every night. Users should copy important files from the client computers to an area on the computer(s) that do the backup.

Considerations for 24 Hour Operations or Very Large Backups

If the data set being backed up is very large, such as a SQL database or a number of large graphics files, it is difficult to back up the data within a reasonable amount of time, given the limitations of existing tape hardware. If Windows NT has locked the file for backup, and it is still locked when the clients start trying to access the file the next morning, the procedure described in the preceding section will not work. If the time available for backups is very small because the files are needed 24 hours a day, you will need a different approach than the one previously described.

You could configure the disks that require daily backup separately from the Windows NT Server system disks by using Redundant Array of Inexpensive Disks (RAID) arrays. For your critical computers, you could implement a software mirror of two separate hardware-controlled RAID arrays. With this configuration, if either a disk or an entire array fails, operations can continue. If a component such as a network interface card, video device, IDE adapter, or power supply fails, it can be easily replaced. If the computer running Windows NT Server itself fails, you should have a spare computer with Windows NT Server already installed to which you can move the data disks.

Note RAID arrays provide disk fault-tolerance completely within hardware that you purchase from a vendor. In RAID arrays, the controller interface handles the creation and regeneration of redundant information that is automatically used if there is a failure in one of the disks in the array. See "Planning a Fault-Tolerant Disk Configuration," presented later in this chapter, for more information about RAID arrays.

One approach to backing up data on a critical computer running Windows NT Server involves logically removing the computer from the network by:

- Disconnecting all users.
- Pausing the Server service by using the Services option on Control Panel.
- Ensuring that all files are closed, so that they can be accessed by the backup procedure.

The following methods are variations on this approach. The first two methods can be used to back up the data once you have removed the computer from the network. In these methods, we call the computer that contains the data to be backed up DATA. The computer that actually does the backup is called TARGET. The third method allows users to continue to access the data on DATA while it is being backed up.

Backup Method 1

Copy the data from DATA to TARGET. When the data are on NTFS volumes, you can use the Scopy program to copy the permissions as well as the data. If you don't need permissions, you can use the Xcopy program to just copy the data. To copy database files, use functions in the database application. If you expect the network to be busy when you will be doing the copy, it is best to dedicate a computer-to-computer link. Once the data are on TARGET, you can bring DATA online and back up the files from TARGET. Copying the data to TARGET minimizes the amount of time that DATA is not available to users.

Note The Scopy program (on the *Windows NT Server Resource Kit* CD) does not support copying only the files that have been modified. The Xcopy program (on the Windows NT Server product CD) does support copying only modified files, but does not copy the file permissions. You need to decide whether Scopy or Xcopy works best for your situation, or whether you need to use a different program to copy the files.

Ideally, you should do a complete file comparison between the data on DATA and TARGET to verify the transferred data. However, comparing the data will take about twice as long as it takes to copy the data. Typical Ethernet or Token Ring transfer speeds are 1 MB per second, if the network is not busy. Use this transfer rate and the total amount of data to transfer to determine the expected transfer time. If you do not have enough time, you will need to use a faster network connection, or a different backup method.

Backup Method 2

Copy the data that you want to back up to another disk or disks on DATA. This process is usually faster than copying to another computer over the network because it occurs within the same computer. You can now bring DATA online. The copy of the data can then be backed up by using a backup device connected to DATA. If you do not want run the backup program on DATA, you can transfer the copy of the data to TARGET, from which you do the backup.

You need to decide whether to do the backup on DATA or TARGET. This decision depends on factors such as:

- Availability of a computer to use as TARGET.
- Your overall backup strategy, which might specify that all backups be performed on a few central computers.
- Whether the processing overhead of doing the backup on DATA is more than the cost and processing overhead associated with transferring the files over the network to TARGET.
- How long you expect the backup to take.

Backup Method 3

There are also third-party utilities that mirror the data onto another computer running Windows NT Server across the network while the files are still being used on DATA. This approach is best when you absolutely cannot take a chance of losing data because a disk or RAID array fails. While there might be some down-time associated with moving data from the mirror computer to the DATA computer after a failure on DATA is fixed, this approach is preferable to losing all data created or changed since the last backup of any kind. Again, performance is better if a network link is dedicated to this task. The other advantage of this approach is that the files being backed up are still available, and there is no downtime required to make copies.

What Data to Back Up

There are different kinds of data that you should back up. This section describes the types of data and why you should back them up.

Registry

The Registry is the most critical set of files on the computer as far as day-to-day operations are concerned. As a user, you would know fairly quickly if a system file is missing or corrupt — the computer would crash or fail to execute the command. The Registry is different. The computer might start, but then hang at the logon screen because all the security settings are missing or some required service did not started. On a computer running Windows NT Server, you need to have a current backup of the Registry. Otherwise, you have to rebuild information about all your users, groups and permissions, and you can never be certain that it is just the way it was before. There are several methods that you can use to back up all or part of the Registry, which are described in Chapter 5, "Preparing for and Performing Recovery." You should develop backup procedures specifically for the Registry to make sure people know who should be doing the backup.

Logon Scripts

Logon scripts are files that can be assigned to user accounts. Each time a user logs on, the assigned logon script is run. You can use logon scripts to configure user working environments by creating network connections and starting applications. Logon scripts are useful when you want to affect the user work environment without managing all aspects of it. A logon script runs automatically whenever a user logs on to a computer running either Windows NT Server or Windows NT Workstation. A logon script is always downloaded from the computer running Windows NT Server that validates a user's logon request.

For users with accounts on Windows NT Server domains that have one or more BDCs, any one of the domain controllers can authorize a user's logon attempt. To ensure that logon scripts always work for users, you should be sure that logon scripts for all user accounts in a domain exist on every PDC and BDC in the domain.

The best way to ensure that logon scripts are always available and consistent is to use the Replicator service. In replication, actual copies of the files are sent across the network to other computers. In the case of logon scripts, you would decide which domain controller should be the sending computer. The receiving computers would be all of the other domain controllers.

You can also replicate information other than Logon scripts, but replication should only be used for critical information that must be available, even if the computer running Windows NT Server it is normally on is unavailable. Replicating files unnecessarily can put a large load on the network and slow it considerably.

Note You should use replication for such critical data as the DHCP and WINS databases. For more information, see Chapter 7, "Using Microsoft DHCP Servers," and Chapter 8, "Managing Microsoft WINS Servers," in the *Windows NT Server Networking Guide.*

See Chapter 3, "Managing User Work Environments," in the *Windows NT Server Concepts and Planning* book, for more information about logon scripts.

Application Programs

Application programs, such as Microsoft Word for Windows, are typically what network users are involved with on a daily basis. You can always reinstall the executable files themselves from the original distribution media, but the down time and lost productivity make this approach less than ideal. Additionally, you might have customized the application programs to suit the needs of your organization. The difficulty of reproducing those settings can be greater than reloading the programs themselves. Fortunately, since the files rarely change, backing up the files are part of your backup procedure insures that the latest version is quickly available, and it does not use a lot of offline storage space. For example, using Ntbackup with the Incremental option insures that any changes made are easily recoverable. Or, you could do a complete backup of the volumes that contain your application programs when you make changes.

User Data

The greatest amount of change on any server is in the users' folders. Users constantly add, modify, or delete files from the computer. You should do daily backups of changes to users' folders.

Some of your users will keep most of the files that they want backed up on file servers. Other users will do most of their work on their own workstation and want that data backed up. Your backup procedures need to cover both situations.

Questions to Consider in Developing Backup Procedures

The critical nature of backups requires complete verification of the entire backup and restore process. Because backup programs vary in their specifics and procedures, this section identifies some of the questions that you should ask, and answer, concerning the process.

Who

These are questions to consider when you are deciding who should be doing certain tasks:

- Who determines what files and computers will be backed up, and how will the policy be published?
- Who is responsible for doing the backup? Is this responsibility formally part of their job description?
- To whom is the success or failure of the backup reported?
- If the designated backup operator(s) are unavailable, who is the alternate?
- How are users notified if the backup fails?
- If the backup takes place unattended, but in a location that has off-shift personnel, are they trained to monitor the backup?
- Should they intervene in case of failure?
- Who do they notify of a problem?
- What is the process they should follow?
- Do they have some way to contact the responsible individual(s)?
- Are these individuals required to be available by telephone or wear pagers?
- If trained personnel are to monitor the backup, how are they scheduled for the next day?
- Is there coverage for their other duties if there are problems with doing the backup, or they need to do a lengthy restore?
- During vacations, sickness, or other absences, who will do the backup?
- If a backup fails because of a hardware problem, who is the contact point with the vendor or manufacturer?
- If the backup fails because of software problems, who do you contact?

When

You also need to consider when and how often the backups should take place:

- If you want to back up as many files as possible, the backup will probably occur after-hours. Will this be true only for full backups or all backups?
- How often do you do full backups and incremental backups?
- At what time should the backup occur, immediately after hours or before business hours early the next day?
- If the backup fails because of hardware problems, is there standby hardware, or can a loaner be obtained from the vendor?
- Will the vendor guarantee availability of a loaner?
- How long will it take to replace the failed hardware?
- Is the vendor's technical support available at all times?
- Is software technical support available at all times?
- Do you have a configuration book that contains information about the computers running Windows NT Server that any technical support person might need?
- If not, how long will it take to create this information when you have a problem?
- What are the policies of the hardware or software vendor concerning fixes and how long they might take?
- How long will it take to retrieve the backups or copies from a local or remote storage area?
- Can the remote copies be obtained at any time or only during business hours?
- How long will it take to do a full restore if the computer totally fails?

Where

Where you do the backups and where you store the backup media are also important questions:

- Is the backup taking place in a secure area?
- If so, how is it monitored?
- Where are the backup tapes stored?
- Is the storage location secure?
- Is it fireproof, waterproof, and otherwise protected from disasters?
- Are the tapes that are stored on site accessible at all times to the people who might need them?
- Are there copies, and where are they stored?
- Is the offsite location secure?

- Is it fireproof, waterproof, and otherwise protected from disasters?
- Is it bonded?
- Will the backup be done to a local tape drive, remotely over the LAN, or remotely over the WAN?
- How is the connection verified before the backup begins?
- How are computers equipped for power outages if operators are not present and backups are taking place?

What

You need to decide what to back up:

- What is the backup plan?
- Are all modified files to be backed up, or will there be a policy regarding specific users, groups, departments, divisions, or company-critical files?
- Will there be disks or volumes on the computer running Windows NT Server that are not backed up?
- Will users be responsible for backup of their individual client systems?
- Will there be a chargeback system for the amount of storage used?

How

You also need to determine how you will do the backups and how to determine if backups work correctly:

- How is the backup process certified?
- Has every option that you expect to use been tested?
- Do the scripts work?
- Do the logs get created and are they correct?
- If the path is long, the filename odd, the file size very large, or the number of files is large, does the backup still work? Can you restore files that have these chacteristics?
- How is the backup started: from the command line, an icon, or by batch?
- Do these methods all work the same?
- If you schedule your backups, do they occur as scheduled?
- Is the tape actually verifying the data?
- Does a test restore work?
- Does the system have to be in a certain condition before the backup starts?
- What is the typical or expected state of the system before, during, and after the backup?
- What is the actual condition?

- Are there any unforeseen behaviors?
- When you make changes to the operating system (such as installing a service pack), or the backup program, do you recertify the backup/restore process?
- If you make hardware changes on the computer, such as installing a new controller or tape drive, or changing the BIOS on the motherboard, do you recertify the backup/restore process?
- How do you certify that you can use your old tapes when you have changed hardware of software involved in the backup?

Storing Data

Some kinds of information should be stored near your computers to be readily available. Some data should be stored off site to be available in the event of a disaster, or for long-term storage. This section discusses both onsite and offsite storage. It assumes that you are using Ntbackup or a similar program to back up your data to a cartridge tape.

Onsite Storage

Some data need to be stored on site and readily available in the event of a failure:

- Daily backups, whether full or incremental. The tapes should always be stored in a fireproof safe or storage cabinet to provide a minimum level of security. Secure storage protects against natural disaster, and theft and sabotage of critical data.
- Copies of the media that you used to install the operating system and application programs should also be stored in a secure storage cabinet. With more software being shipped on CD-ROM, it is not easy to just make a copy like you do with floppy disks. If more than one copy of the program is purchased, an extra copy can be stored off site. If only one copy is available, then it should be backed up to an individual tape, and labeled as a backup of the application. If you need to reinstall the software, you can restore from tape onto a computer running Windows NT Server. You can then run the application program setup over the network. (The software then needs to be deleted from the server.)

Offsite Storage

The most important data should always be stored off site. There are many companies that specialize in secure data storage. These companies lease space in underground vaults that remain impervious to just about everything. Smaller companies may find the cost and logistics of this degree of protection to be too great. Alternative solutions can be as simple a safe deposit box at a local bank or a fireproof safe at home or another location.

You should store the following types of information off site:

- A full backup of the entire system should be done on a weekly basis and be stored off site.
- The original of all installed programs should be stored off site. Copies should be kept on site.
- Anything that an insurance company is going to need to facilitate a claim should be stored off site, such as copies of purchase orders or receipts for computer hardware and software. See your business insurance agent for the specific information they will need.
- A copy of any information required to get network hardware reinstalled or reconfigured. This information is usually very hardware specific. Check with your phone company or other supplier of data transmission lines for information that will help them get your service back online as quickly as possible.

Planning a Fault-Tolerant Disk Configuration

A technology called Redundant Array of Independent Disks (RAID) minimizes loss of data caused by problems with accessing data on a hard disk. RAID is a fault-tolerant disk configuration in which part of the physical storage capacity contains redundant information about data stored on the disks. The redundant information enables regeneration of the data if one of the disks or the access path to it fails, or a sector on the disk cannot be read.

Some vendors sell disk subsystems that implement RAID technology completely within the hardware. Some of these hardware implementations support hot swapping of disks, which enables you to replace a failed disk while the computer is still running Windows NT Server. RAID arrays are not described in this book.

In this chapter, and other chapters in the "Reliability and Recoverability" part of this book, the term *RAID array* refers to a fault-tolerant disk subsystem where all of the fault tolerance is implemented in the hardware. For information about RAID arrays that are compatible with Windows NT, see the Windows NT Hardware Compatibility List (HCL). Chapter 8, "General Troubleshooting," contains more information about the Windows NT HCL.

Windows NT Server provides software support for two fault-tolerant disk configurations: mirror sets and stripe sets with parity. You use Disk Administrator to configure mirror sets and stripe sets with parity, and to reconstruct the volume when there has been a failure.

Windows NT Server and Windows NT Workstation have software support for stripe sets, which you configure by using Disk Administrator. Although stripe sets are not fault-tolerant, they are described in this section because:

- Stripe sets are the basis for stripe sets with parity.
- Stripe sets improve I/O performance.
- Windows NT Workstation and Windows NT Server need the same kind of information, both on disk and in the Registry, to identify and access stripe sets, mirror sets, and stripe sets with parity.

All RAID disk configurations:

- *Regenerate* data to satisfy a read request when a disk or a path to a disk has failed. Regeneration involves reading data from other disks in the volume. The regeneration occurs automatically.
- *Reconstruct* the missing data onto the new disk when you have replaced the failed disk (or the path to it). When you have implemented a fault-tolerant volume by using Disk Administrator, you also have to use Disk Administrator to begin the reconstruction of the data. If you are using a RAID array, see the vendor's documentation for information about reconstructing data.

For complete information about RAID, RAID terminology, and disk arrays, see *The RAIDbook, A Source Book for Disk Array Technology*. The RAID Advisory Board in St. Peter, MN, publishes this book. The June 1995 edition is the basis for RAID information in this chapter.

Chapter 7, "Protecting Data," in Windows NT Server *Concepts and Planning*, contains more information about using stripe sets, mirror sets, and stripe sets with parity on Windows NT Server.

Chapter 5, "Preparing for and Performing Recovery," in this book, describes configuring, and recovering from failures of, mirror sets and stripe sets with parity.

Stripe Set

Stripe sets are composed of *strips* of equal size on each disk in the volume. You create a stripe set from equal sized, unallocated areas on two to 32 physical disks. For Windows NT Workstation and Windows NT Server, the size of the strips is 64K.

Conceptually, a stripe set is similar to a table in a document, where a disk is a column and a strip is one of the entries in the table. A stripe is all of the entries in one row. The next table shows how a stripe set is similar to a table. It also shows the order in which data are written to a stripe set.

Table 4.1 Stripe set

	Disk 1	Disk 2	Disk 3	Disk 4
Stripe 1	1	2	3	4
Stripe 2	5	6	7	8
Stripe 3	9	10	11	12
Stripe 4	13	14	15	16
Stripe 5	17	18	19	20

The next figure shows a stripe set using four disks. A strip is one of the blocks on one of the disks. Like the preceding table, stripe 1 consists of the four strips that are the first block on each of the four disks. Stripe 5 is made up of the strips that are the last block on each disk.

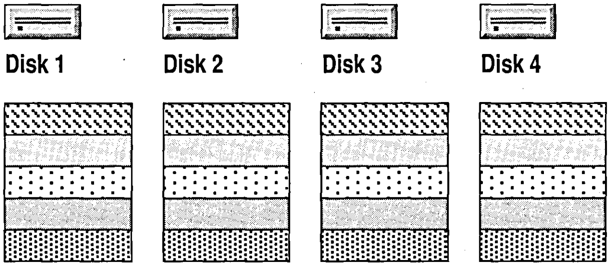


Figure 4.1. Stripe set

When you write data to a stripe set, the data are written *across* the strips in the volume. Thus, using the Table 4.1 and Figure 4.1 as the example, if you wrote a file that was 325K long, it would occupy the following areas:

- 64K on strip 1 of disk 1.
- 64K on strip 1 of disk 2.
- 64K on strip 1 of disk 3.
- 64K on strip 1 of disk 4
- 64K on strip 2 of disk 1.
- 5K on strip 2 of disk 2.

The physical disks in a stripe set do not need to be identical, but there must be unpartitioned areas available on each disk that you want to include in the volume. You cannot add disks to a stripe set if you need to increase the size of the volume later, because Windows NT would have to move the data around on the stripe set. If you want to change the size of a stripe set, you have to:

- Back up the data.
- Delete the stripe set by using Disk Administrator.
- Create a new stripe set, also by using Disk Administrator.
- Restore the data to the new stripe set.

Stripe sets do not contain any redundant information. Therefore, the cost per MB for a stripe set is identical to that for the same amount of storage configured from a contiguous area on a single disk. Although the data are spread across multiple disks, there is no fault tolerance. When any disk fails, the whole stripe set fails, and no data can be recovered. The reliability for the stripe set is worse than the least reliable disk in the set.

You create and use a stripe set for performance reasons. Access to the data on a stripe set is usually faster than access to the same data would be on a single disk, because the I/O is spread across more than one disk. Therefore, Windows NT can be doing seeks on more than one disk at the same time, and can even have simultaneous reads or writes occurring.

A stripe set works well in the following environments:

- When users need rapid read access to large databases or other data structures.
- Storing program images, DLLs or run-time libraries for rapid loading. Operating systems that use memory mapped images, such as Windows NT, can benefit from using stripe sets.
- Collecting data from external sources at very high transfer rates. This is especially useful when the collection is done asynchronously.
- Multiple independent applications require access to data stored on the stripe set. When the operating system supports asynchronous multithreading, this insures disk reads and writes can be load balanced.

Stripe sets are not well suited for the following situations:

- When programs make requests for small amounts of sequentially located data. It would probably be more efficient to use one volume on a single disk for this type of data, because other users' data requests would not be impacted as much. For example, if a program requests 8K at a time, it takes eight separate I/O requests to read or write all of the data in a 64K strip, which is not a very good use of this storage mechanism.
- When programs make synchronous random requests for small amounts of data. This causes I/O bottlenecks because each request requires a separate seek operation. 16-bit single-threaded programs are very prone to this problem.

Mirror Set

A mirror set provides an identical twin for the selected partition. All data written to the mirror set are written to both partitions, which results in disk space utilization of only 50 percent.

Creating a mirror set is similar to making a copy of a document by using a copy machine. The original partition is like the original of the document, and the shadow partition is the copy. Unlike a copy machine, however, Windows NT continually updates both the original and shadow partitions when any changes are made to the mirror set.

Note In this chapter and other chapters in the "Reliability and Recoverability" part of this book, the terms *original disk* and *original partition* refers to the original partition, and *shadow disk* and *shadow partition* refers to the disk or partition that contains the copy.

Figure 4.2 shows a mirror set. To a user, only one read or write occurs to satisfy a request for data. FtDisk, the Windows NT fault-tolerant disk driver, creates separate I/O requests for each of the disks.

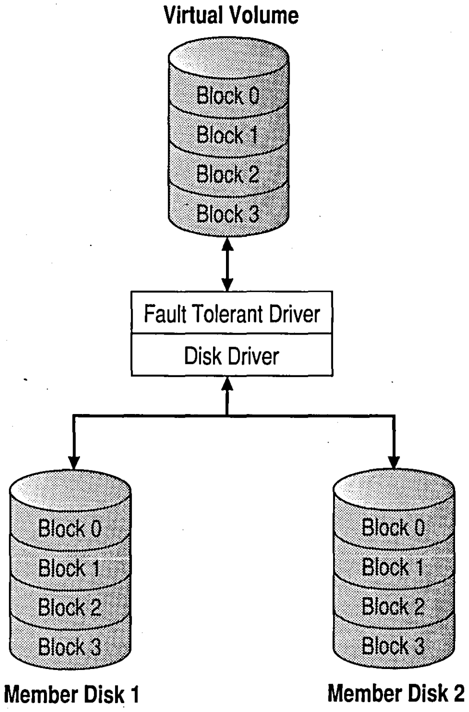


Figure 4.2. Mirror set

If there is a read failure on one of the disks, FtDisk reads the data from the other disk in the mirror set. If there is a write failure on one of the disks in the mirror set, FtDisk uses the remaining disk for all accesses.

Because dual-write operations can degrade system performance, many mirror set implementations use duplexing, where each disk in the mirror set has its own disk controller.

Any primary partition can be mirrored, including the system and boot partitions. The disk that you select for the shadow partition does not need to be identical to the original disk in size, or number of tracks and cylinders. This means that you do not have to replace a failed disk with an identical model. The unpartitioned area that you select for the shadow partition cannot be smaller than the original partition. If the area that you select for shadow partition is larger than the original, the extra space on the shadow disk can be configured as another partition (if there are fewer than four partitions on the disk).

Note If you configure your system partition as a mirror set, you should use identical disks for the mirror set, and use the same disk geometry. See Chapter 5, "Preparing for and Performing Recovery," for details about configuring mirror sets for these partitions.

When compared to stripe sets with parity (described later in this chapter), a mirror set implementation:

- Has a lower entry cost (because it requires only two disks, whereas a stripe set with parity requires three or more disks).
- Requires less system memory.
- Provides the best overall performance.
- Does not show performance degradation during a failure.
- Has a higher cost-per-megabyte.

You can mirror the boot partition, which significantly reduces the amount of time needed to get your Windows NT Server back up if there is a problem with the hard disk containing your operating system.

Note The term system partition refers to the disk partition containing hardware-specific files needed to start Windows NT (such as the x86-based files NTLDR and Boot.ini, or the RISC-based files Osloader.exe and Hal.dll). The boot partition contains the operating system files and support files. The boot partition and the system partition can be the same partition.

A mirror set works well for the following applications:

- When extremely high data reliability is required. A duplexed mirror set has the best data reliability because the entire I/O subsystem is duplicated.
- When the best possible performance is required. Performance is excellent for read-intensive applications, with only a slight penalty for disk writes.
- When simplicity is important. Mirror sets are simple to understand and easy to set up.

You might not want to use a mirror set if cost is a critical factor. Mirror sets are the most expensive solution on a cost per unit of data storage basis.

Advantages to Using Mirror Sets

Disk read operations are more efficient than on a single partition. FtDisk, the fault tolerance disk driver, has the capacity to load balance read operations across the physical disks. With current SCSI technology, two disk read operations can be done simultaneously. In some cases, a disk read can be done in half the time it would take on a single partition.

Recovery from a disk failure is very rapid. Mirror sets offer the fastest data recovery, with the least impact on system performance, because the shadow partition contains all of the data. There is no data rebuilding needed to restore the system. When you configure your boot partition on a mirror set, you do not have to reinstall the Windows NT Server to be able to restart the computer.

It is not necessary to use identical physical disks or to have the same partitions on each disk, although you should use identical disks if you put your system partition on a mirror set. All FtDisk requires is sufficient unused space on the second disk to create the shadow partition.

You can mirror the system or the boot partition. You cannot use a stripe set with parity for either of these partitions.

Disadvantages of Mirror Sets

Disk write operations are less efficient. Because data must be written to both disks, there is a slight performance penalty. However, since disk writes are done asynchronously, the penalty is not 100 percent. In most situations, a user-mode application would not be affected by the extra disk update.

Mirror sets are the least efficient in terms of space utilization. Because the data are duplicated, the space requirements for a mirror set are higher than for a stripe set with parity.

When you create a mirror set for the system or boot partition, you should create and test a Windows NT startup floppy disk in order to be able restart the computer. In the case of RISC-based computers, you need to create an alternate boot selection to be able to use the Windows NT startup floppy disk. Otherwise, the time to restart the computer after a failure increases because you have to create the boot selection.

Creating a mirror set of a boot or system partition does not do a sector by sector duplication of the disk. FtDisk does not copy either the Master Boot Record on track 0 or the Partition Boot Sector, which is the first sector of the partition. Some systems might save information in other parts of track 0, and this will not be duplicated either. Data corruption in these areas can make it much more difficult to recover from a failure of your original disk. If both disks are affected, it can be extremely difficult or impossible to recover the data.

For information on recovering from the failure of a mirror set, see Chapter 5, "Preparing for and Performing Recovery."

Stripe Set With Parity

A stripe set with parity adds parity information to a stripe set configuration. A stripe set with parity dedicates the equivalent of one disk's space in the stripe set with parity for storing the parity strips, but distributes the parity strips across all the disks in the group. The data and parity information are arranged on the volume so that they are always on different disks.

Table 4.2 shows the order in which data are written to a stripe set with parity that consists of five stripes on five disks.

Table 4.2 Stripe set with parity

	Disk 1	Disk 2	Disk 3	Disk 4	Disk 5
Stripe 1	parity 1	1	2	3	4
Stripe 2	5	parity 2	6	7	8
Stripe 3	9	10	parity 3	11	12
Stripe 4	13	14	15	parity 4	16
Stripe 5	17	18	19	20	parity 5

In Figure 4.3, the first block on disk 1 is the parity strip for the four data strips in stripe 1. In stripe 2, the parity strip is on disk 2, and so on. Figure 4.3 is another representation of the information shown in Table 4.2.

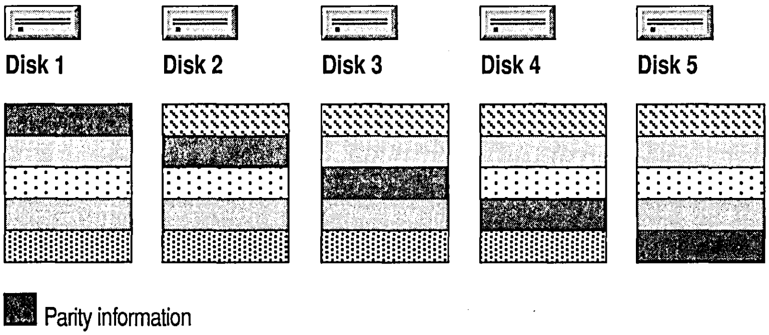
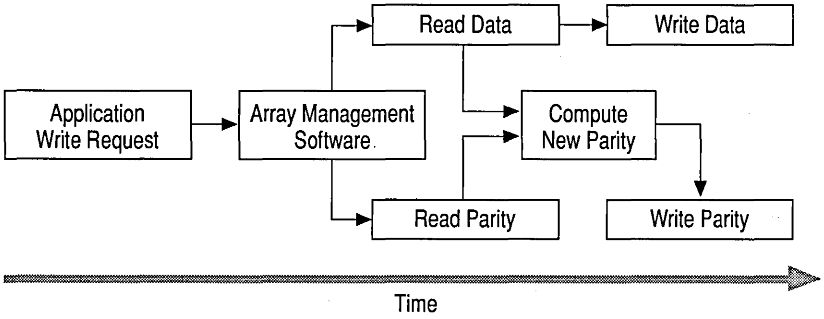


Figure 4.3. Stripe set with parity

The parity strip is the exclusive OR (XOR) of all the data values for the data strips in the stripe. If no disks in the stripe set with parity have failed, the new parity for a write can be calculated without having to read the corresponding strips from the other data disks. Thus, only two disks are involved in a write operation: the target data disk and the disk that contains the parity strip. The next figure shows the steps that are involved in writing data to a stripe set with parity.



When implementing a stripe set with parity, there must be at least three disks and no more than 32 disks in the set. The physical disks do not need to be identical. However, there must be equal size blocks of unpartitioned space available on each physical disk in the set. The disks can be on the same or different controllers. As with stripe sets, you cannot add disks to a stripe set with parity if you need to increase the size of the volume later.

If one of the disks in a stripe set with parity fails, none of the data are lost. When a read operation requires data from the failed disk, the system reads all of the remaining good data strips in the stripe and the parity strip. Each data strip is subtracted (with XOR) from the parity strip; the order isn't important. The result is the missing data strip.

When the system needs to write a data strip to a disk that has failed, it reads the other data strips and the parity strip and backs them out of the parity strip, leaving the missing data strip. The modifications needed to the parity strip can now be calculated and made. Because the data strip is bad, it is not written; only the parity strip is written.

There is no effect on a read operation when the disk that failed contains a parity strip. (The parity strip isn't needed for a read, unless there is a failure in a data strip.) When the failed disk contains a parity strip, the system does not compute or write the parity strip when there is a change in a data strip.

A stripe set with parity works well in the following situations

- In large databases where reads occur much more often than writes. Performance degrades as the percentage of write operations increases. Database applications that do random reads work well with the built-in load balancing of a stripe set with parity.
- Where a high degree of fault tolerance is required without the expense of a mirror set. A stripe set with parity is several times more efficient with storage than a mirror set when larger numbers of disks are used. The space required for storing the parity information is equivalent to 1/Number of disks, so a 10 disk array uses 1/10 of its capacity for parity information.

A stripe set with parity is not well suited for:

- Applications that require high-speed data collection from a process. This type of application requires continuous high-speed disk writes, which do not work well with the asymmetrical I/O balance inherent in stripes set with parity and the extra I/Os required to write the parity strip.
- Database applications in which records are continually updated. Financial applications where balances are frequently updated are a typical example.
- Applications that require large sequential data transfers. These types of data transfer can prevent effective I/O load balancing.

Advantages of Stripe Sets With Parity

Disk read operations can occur simultaneously. All disks in the array can be in use at the same time.

Cost is lower than for a mirror set. The utilization increases as the number of disks in the array increases.

Disadvantages of Stripe Sets With Parity

Neither the system partition or boot partition can be on a stripe set with parity.

Write operations are substantially slower than for a single disk because the software has to read the old data strip, the old parity strip, and compute the new parity strip before writing it.

If a disk that is part of a stripe set with parity fails, read operations for data strips on that disk are substantially slower than for a single disk. The software has to read all of the other disks in the set to calculate the data.

A stripe set with parity requires more system memory than a mirror set.

Note The minimum RAM on a RISC-based computer is 16 MB. On an x86-based computer, 12 MB suffices, but 16 MB is recommended. You might want to add memory when you configure disks as stripe set with parity.

Fault-tolerant Hardware Versus Software

On heavily-loaded computers running Windows NT Server, there might be advantages to using a hardware solution for mirror sets or stripes set with parity. Or you might want to select a different type of RAID array. There are many factors involved, which depend on the specific site, hardware being used, and the load on the computer running Windows NT Server.

Here are some points to consider when deciding whether to implement fault tolerance in hardware or software.

- Fault-tolerant software is available only on Windows NT Server.
- Hardware fault tolerance is faster.
- Software fault tolerance is less expensive.
- A hardware fault tolerance solution might lock you into a single vendor.
- In a hardware fault tolerance implementation, some vendors support hot-swappable disks when there is a disk failure.

Regardless of whether you implement fault tolerance by using hardware or software, implementing fault tolerance does not reduce the need for backups.

Guidelines for Configuring Your Mass Storage

There is no optimal mass storage configuration for your Windows NT computer. The tradeoffs involved in deciding your configuration depend upon such variables as cost, performance, and how much time you can afford to take to completely rebuild a disk. This section discusses why you might want to use mirror sets, stripe sets with parity, or a combination of the two fault-tolerant software techniques on your Windows NT Server.

In general, you only need to use fault-tolerant configurations for information that you must have readily available in case of hardware failure or unrecoverable disk errors. You do not need to have your page file on a fault-tolerant volume, and you definitely should not have the page file on a stripe set with parity because of potential performance impacts.

Note If you configure your computer running Windows NT Server to write a memory dump file each time it generates a Kernel STOP error, your page file must be located on the boot partition. If you configure your computer this way, and create a mirror set of your boot partition, the page file will be mirrored.

You might not need to configure application programs on a fault-tolerant volume. If you are considering having application programs and other common files on more than one computer running Windows NT Server, you can use replication to keep them all consistent and to provide the redundancy. If you have applications on a single computer running Windows NT Server, you only need to configure them on a fault-tolerant volume if you cannot tolerate them being unavailable for the amount of time it takes you to restore them from a backup. Be sure to back up the application program volume any time you install a new application or change default settings for an application.

You can put the application programs on a stripe set to get the fastest I/O performance for reading data.

If space is a consideration, you can format your application program volume with the NTFS file system (or convert a FAT volume to an NTFS volume), and use NTFS compression for folders and files on the volume. See Chapter 18, "Choosing a File System," in the *Windows NT Workstation Resource Guide* for details about NTFS compression.

Guidelines for Using Mirror Sets and Stripe Sets with Parity

There are software and hardware constraints to consider, as well as the cost and reliability of the system. Each potential vendor should have design guidelines for their system, whether you are looking for a RAID array or want to use the fault-tolerant features in Windows NT Server.

There are configurations and situations in which fault tolerance might not work as you would expect. The first thing you should check is whether all of the disk hardware is on the Windows NT Hardware Compatibility List (HCL). If any equipment is not included on the HCL, it might not work well. Chapter 8, "General Troubleshooting," contains more information about the HCL.

If you can, use identical disks. Although this is not a requirement, there are several advantages in using identical hardware for fault-tolerant configurations:

- The performance of the disks is the same. Faster disks do not have to wait for slower disks.
- If you want to configure the entire disk for a mirror set or stripe set with parity, the capacity is the same.
- There are fewer potential problems with configuration and compatibility.

Have a backup disk available and ready for use. When purchasing the disks, it is a good idea to purchase an extra disk for use as a backup. An identical disk guarantees that in the event of a hardware failure, the backup disk is compatible, and will not degrade the system performance. All you need to do is install the new disk. (For SCSI disks, you have to configure the SCSI ID.)

Have a backup controller available, or configure your disks with duplexed controllers. The system can continue operating when a disk fails, but not when a controller fails. While there is no way to avoid system downtime when a controller fails, downtime can be minimized by having a pre-configured controller available. If the replacement controller is not the same as the original, you have to install a new driver, and your configuration becomes more complex.

Note If you have configured your system partition on a mirror set, you should have the same make and model of controller for your backup, and be sure to use the same translation for both the original and shadow partition.

Configuring Mirror Sets

To a large extent, how your configure your mirror sets depend on the number of disks and controllers that you want to have on the computer running Windows NT Server. Having your boot partition on a disk (and controller) that does not contain data sets gives better performance. There are also other cost considerations that are related to overall reliability, such as your UPS strategy.

The decision to mirror the system or boot partition depends on how fast you want to be able to get your computer back up following a failure to the disk that contains these files. To some extent, it depends on whether you are using a x86-based computer or a RISC-based computer — x86-based computers experience most of the problems with sectors corrupted by viruses. These problems mean that you cannot start your computer from disk, although you can usually start it by using the Windows NT startup floppy disk. See Chapter 5, "Preparing for and Performing Recovery," for information about the Windows NT startup floppy disk.

When you use a mirror set for your system or boot partitions, these are some configurations that might cause problems:

- Multichannel controllers do not always work for duplexing. It might not be possible to start from the second channel of a dual-channel controller by using the Windows NT startup floppy disk, which means you cannot restart your computer at all. In this situation, you would have to use the hardware configuration program to change the boot channel, or physically move the disks to the other channel.
- Be sure to use the same disk translation for the disks in a mirror set of the system partition, and to format both disks using the same controller. Chapter 5, "Preparing for and Performing Recovery," contains more information about avoiding problems with incompatible geometry.

Configuring Stripe Sets With Parity

For a stripe set with parity, you should buy disks on the basis of cost per megabyte and performance. Since the space used for storing parity information is 1/n, storage efficiency increase with a larger number of disks. If you use high performance SCSI controllers, an array of six 2 GB disks provides more efficient storage than four 3 GB disks. If you have to replace a disk, you can rebuild a smaller disk faster, so the system returns to maximum performance is less time.

Keep the stripe set with parity on a different controller and disk than your system and boot partition. Using separate controllers improves performance, and can make it faster and easier to recover from a hardware failure. Remember that you cannot configure your system partition or your boot partition on a stripe set with parity.

If the computer itself or the disk containing the boot partition fails (and the boot partition is not mirrored), it might be faster to move the stripe set with parity to a different computer. Chapter 7, "Disk, File System, and Backup Utilities" contains information about moving disks in the section "Using FT Edit to Update the Registry."

Sample Configuration With Mirror Set and Stripe Set With Parity

If you want most of your data to be on fault-tolerant volumes, you might still want to use a small, non-fault-tolerant disk in your configuration. This disk could contain the system and boot partitions, and the page file. Ideally, the page file can be on its own partition to keep it from becoming fragmented. You should also have enough space on this disk for a temporary file large enough to use to install a service pack.

If you have configured your computer to write a memory dump when a Kernel STOP error occurs, Windows NT creates a page file the size of the RAM in your computer plus 1 MB. This page file must be on the boot partition. Therefore, you need to allow for a file this size if you are configured for memory dumps.

You can also create a mirror set of this small disk. You do not need to include your page file in the mirror set. You could have a separate partition for the page file on each of the disks in the mirror set. However, if you create a mirror set of your boot partition, and have configured the computer to write a memory dump file, there will be a page file on each partition in the mirror set. For more information about page files and memory dumps, see Windows NT Help.

Note Small in this case must be large enough for the system partition (5-10 MB), the boot partition (150-175 MB), the page file (RAM size + 1 MB), and room for a temporary folder (10-15 MB). This small disk should be no smaller than 250 MB, and 300 MB is a better size.

You should not mix the boot partition, stripe sets, stripe sets with parity, and mirror sets on the same physical disks. Doing so can create problems when a disk failure occurs. For example, you could configure a system with three large physical disks, as follows:

- The first disk has the system partition and boot partition using half the disk. A stripe set with parity occupies the remaining space on the disk.
- Half of disks two and three contain the other partitions of the stripe set with parity.

If the disk with the system and boot partitions fails, the operating system cannot be started. The Registry, with the information about the stripe set with parity, cannot be accessed. You have to install a replacement disk, install Windows NT Server, and restore the Registry. Expect to take at least one hour before you can restart the computer.

A better solution would be to use five smaller disks, as follows:

- Configure the first two disks as a mirror set containing the system and boot partitions. Be sure the page file is on the mirror set and not the stripe set with parity.
- Create a stripe set with parity on the other three disks.

If the original disk of the mirror sets fails, you can restart the system from the shadow disk. Downtime should be less than five minutes. Chapter 5, "Preparing for and Performing Recovery," describes what you need to do to restart from the shadow disk.

CHAPTER 5

Preparing for and Performing Recovery

This chapter describes what you can do to:

- Reduce the possibility of problems.
- Recover from problems when they occur.

There is no way to make a computer running Windows NT failure proof. You can only make the computer more failure resistant. A memory module, cabling, or controller failure can corrupt data on any disk. In this event, the only option is to restore from a tape or a backup server that contains a copy of the data.

These are some things that you can do to make it easier to recover from problems:

- Develop plans and procedures for recovering from failures before you have one.
- Create and test floppy disks that enable you to restart the computer when you are having trouble starting from the system partition.
- Maintain software configuration information for your computers running Windows NT. At a minimum, keep track of the version of the operating system installed on each computer, including service packs and hotfixes.
- Record the hardware configuration of each computer running Windows NT, especially the disk configurations.

The topics covered in this chapter are:

- Maintaining configuration and other forms of information.
- Understanding ARC pathnames.
- Creating floppy disks to use to start the computer.
- Training and testing.
- Restoring disk information.
- Configuring and using mirror sets and stripe sets with parity.
- Using an uninterruptable power supply.

This chapter also describes using utilities on the *Windows NT Server Resource Kit* CD and the *Windows NT Server* CD to back up and restore critical data. Planning what utilities to use is as important as knowing how to use them. There is more information about these utilities in Chapter 7, "Disk, File System, and Backup Utilities."

Chapter 4, "Planning a Reliable Configuration," discusses planning what to do before failures occur, such as:

- Deciding what hardware and software to procure.
- Developing your plan for doing backups.
- Developing training plans, disaster plans, and contingency plans.

These are some points to consider:

- Investigate the various utilities for doing a task, and decide which one(s) to use for which situations.
- When you have decided which utilities you want to use, prepare your floppy disks.
- Know how to use the utilities that you decide to use.
- Practice recovering from the common problems described in this chapter and in Chapter 6, "Troubleshooting Startup and Disk Problems."

Maintaining Configuration and Essential System Information

Hardware failures, power failures, and human errors can corrupt information that your computer needs to start Windows NT.

Recovery is easier if:

- You know the configuration of each computer and its history.
- You back up critical system files every time you make certain changes to your Windows NT configuration.

On computers running Windows NT Server, you can purchase disk subsystems that maintain redundant information on the disks. The redundant information is either parity information, or a complete, separate copy of the data. Having redundant information makes the disk subsystems fault tolerant, meaning that you can continue to access the data when one disk in a fault-tolerant configuration fails. This fault-tolerant technology is called RAID, for Redundant Array of Inexpensive Disks. A disk subsystem that implements fault tolerance using RAID technology is also called a RAID array.

Windows NT Server implements two of the fault-tolerant configurations in software: mirror sets, and stripe sets with parity. For a description of mirror sets and stripe sets with parity, see Chapter 4, "Planning a Reliable Configuration."

For complete information about RAID, RAID terminology, and RAID arrays, see *The RAIDbook, A Source Book for Disk Array Technology*. The RAID Advisory Board in St. Peter, MN, publishes this book. The June 1995 edition is the basis for RAID information in this chapter. *The RAIDbook, A Source Book for Disk Array Technology* also contains information about parity.

Keeping a Log Book

You should have a log book for every computer, which contains information about the computer's configuration. Having current information makes it easier to rebuild your computer in the event of a serious system crash. This information also helps product support personnel to troubleshoot problems. The Registry contains information about your disk configuration in the subkeys HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap\AtDisk and Scsi.

The type of information to keep includes:

- Computer type, model number, and serial number.
- Computer BIOS manufacturer and revision level (x86-based computers) or firmware revision level (RISC-based computers).
- CMOS information for x86-based computers.
- NVRAM information on RISC-based computers.
- Hardware configuration information, including IRQs, DMA addresses, I/O ports, and similar information. If the computer has an EISA bus, the EISA configuration and its associated CFG files should be backed up.
- SCSI controller model number, and the BIOS firmware revision level.
- Jumper settings for all peripheral devices.
- Complete map of the SCSI subsystem, including:
  - SCSI configuration information from the SCSI setup program.
  - Which devices are terminated and how they are terminated.
  - The SCSI ID and physical location on the chain of every SCSI device.

- Which versions of Windows NT are installed, and the partitions on which they are installed.
- Details of any device drivers or other system level software that did not come in the Windows NT retail package. This software would include such things as a Network File System (NFS) provider, network protocol, or network management software.
- Troubleshooting history for any system failures or Kernel STOP errors (blue screens). This information should include:
  - The time and date the problem occurred.
  - Any error messages, or events posted to the event log.
  - Any troubleshooting done and the outcome.
- Partition information, such as the size of the partitions and the file system used for each one. You can use the DiskMap program, described in Chapter 7, "Disk, File System, and Backup Utilities," to print a map of each of your disks by redirecting the output to a file or a printer.

Backing up the Registry

All of the Windows NT internal configuration information is stored in the Registry. It is critical to have up-to-date, reliable backups of this information. There are several ways to back up the all or part of the Registry, and you should have redundancy in this area.

If you have a backup device installed on your computer, you can include the Registry in normal backups. You have to specifically select this option in the Windows NT tape backup program Ntbackup.exe. Third-party Windows NT-compatible backup software might also require that you take specific action to back up the Registry. Be sure to test your Registry backup and restore procedures.

Note Windows NT Backup can only back up the Registry on the local system. You cannot use Windows NT Backup to back up a computer's Registry over the network.

After any change in the configuration of the operating system, including adding new software, you should use the Repair Disk program, Rdisk.exe, to update files in the *%systemroot%*\Repair folder, and create a new Emergency Repair Disk. For information about using the Repair Disk program, see "Creating an Emergency Repair Disk," presented later in this chapter. It is a good idea to make a copy of the Emergency Repair Disk and store it in a separate location, perhaps off site.

You can use the Windows NT Registry Editor, Regedt32.exe, to save and restore Registry keys.

The *Windows NT Server Resource Kit* CD contains utilities that enable you to back up and restore all or part of the Registry. For information about the Registry Backup (Regback.exe) and Registry Restore (Regrest.exe) utilities, see Rktools.hlp.

Saving the SYSTEM Key

The HKEY_LOCAL_MACHINE\SYSTEM key contains configuration information that Windows NT uses during startup. Even though this key is copied to the Emergency Repair Disk when you use Repair Disk program, there are times that you might want to restore this key from another disk. For more information, see "Restoring the SYSTEM Key," presented later in this chapter.

You can save the SYSTEM key by using either Disk Administrator or the Windows NT Registry Editor (Regedt32.exe). The primary difference between these two methods is that the Disk Administrator always saves the key to a floppy disk, whereas you can specify any disk by using the Registry Editor.

Note The *%systemroot%*\System32\Config\System file is usually larger than the SYSTEM key saved by using Disk Administrator or Registry Editor. The smaller size results because the two programs compact the file before copying it to eliminate internal fragmentation and holes.

▷ **To use Disk Administrator to save the SYSTEM key**

1. On the **Partition** menu, click **Configuration**.
2. On the **Configuration** menu, click **Save**. A message is displayed describing what will be saved and where you should save it.
3. Insert any floppy disk with enough unused space to hold the configuration information (about 512K). Using the Windows NT startup floppy disk is highly recommended.
4. Click **OK** to write the data to the floppy disk.

▷ **To use Registry Editor to save the SYSTEM key**

1. Run Registry Editor (Regedt32.exe).
2. Click the HKEY_LOCAL_MACHINE\SYSTEM key. On the **Registry** menu, click **Save key**.
3. Enter the path to the file where you want to save the key.

Saving the DISK Subkey

The HKEY_LOCAL_MACHINE\SYSTEM\DISK subkey in the Registry contains configuration information about currently defined drive letters, volume sets, stripe sets, stripe sets with parity, and mirror sets, as well as CD-ROM mappings and drive mappings. Any time that you make changes to your disk configuration, Windows NT updates the DISK subkey. You should save this key whenever you make changes that affect it, such as:

- Creating or deleting a disk partition.
- Changing a drive letter.
- Creating or deleting a volume set, stripe set, mirror set, or stripe set with parity.

Having a backup is useful in situations when you do not want to restore the entire Registry or the entire SYSTEM key, such as the following:

- You want to move a volume set, stripe set, or stripe set with parity to another computer because of a hardware failure.
- Configuration information about a volume has been corrupted.
- You need to replace hardware.

For example, if you have changed a SCSI controller, restoring the entire SYSTEM key might make it difficult to start the computer. Restoring the DISK subkey enables you to have current disk configuration information without changing anything else in the Registry.

Note The DISK subkey does not exist the first time that you start Windows NT. Disk Administrator creates the key the first time that you run it.

▷ **To use the Registry Editor to save the DISK subkey**

1. Run Registry Editor (Regedt32.exe).
2. Click the HKEY_LOCAL_MACHINE\SYSTEM\DISK subkey. On the **Registry** menu, click **Save key**.
3. Enter the path to the file where you want to save the key.

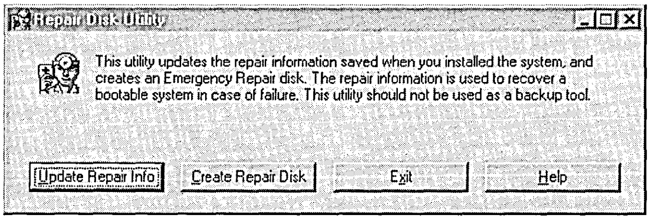
Creating an Emergency Repair Disk

When you install Windows NT, Setup creates the Registry information in *%systemroot%*\System32\Config. For recovery purposes, Setup also creates a *%systemroot%*\Repair folder that contains the following files:

File Name	Contents
Autoexec.nt	Copy of *%systemroot%*\System32\Autoexec.nt, which is used to initialize the MS-DOS environment.
Config.nt	Copy of *%systemroot%*\System32\Config.nt, which is used to initialize the MS-DOS environment.
Default._	Registry key HKEY_USERS\DEFAULT, compressed.
Ntuser.DA_	Compressed version of %systemroot%\Profiles\Default User\Ntuser.dat. The repair process uses Ntuser.da_ if this one needs to be repaired.
Sam._	Registry key HKEY_LOCAL_MACHINE\SAM, compressed.
Security._	Registry key HKEY_LOCAL_MACHINE\SECURITY, compressed.
Setup.log	Log of which files were installed, and cyclic redundancy check (CRC) information for use during the repair process. This file is Read Only, System, and Hidden, so it will not be visible unless you have configured My Computer or Windows NT Explorer to show all files.
Software._	Registry key HKEY_LOCAL_MACHINE\SOFTWARE, compressed.
System._	Registry key HKEY_LOCAL_MACHINE\SYSTEM, compressed.

During installation, Windows NT Setup asks whether you want to create an Emergency Repair Disk. You should choose **Yes**. When you create an Emergency Repair Disk during installation, or later by using the Repair Disk program (Rdisk.exe), the files are copied from *%systemroot%*\Repair to the floppy disk. You can also use the Emergency Repair Disk to repair a corrupt Partition Boot Sector for the system partition.

When you run the Repair Disk program, you can update the *%systemroot%*\Repair\Software and *%systemroot%*\Repair\System files with the current information in *%systemroot%*\System32\Config by using the Update Repair Info button.


This utility updates the repair information saved when you installed the system, and creates an Emergency Repair disk. The repair information is used to recover a bootable system in case of failure. This utility should not be used as a backup tool.

The Repair Disk program does not update the Default, Sam and Security files if you run the program from Windows NT Explorer or My Computer. To update all the files, you can run **rdisk** from the command prompt by entering:

```
rdisk /s
```

You can also update all of the files by clicking the Start button, clicking Run, and entering:

```
rdisk /s
```

Using the /s switch forces the Repair Disk program to update all of the Registry keys in the *%systemroot%*\Repair folder.

There are several points to consider about maintaining and using your Emergency Repair Disk:

- If you did not use the **/s** switch before you create the Emergency Repair Disk, and you use the Repair process to replace the SYSTEM Registry keys, all passwords in the system return to the passwords in effect at the time you last updated the Sam and Security files in *%systemroot%*\Repair. If you have more recently backed up these keys by using Windows NT Backup, Regback.exe, or Regedt32.exe, you can restore more current information.
- If you did not update the Emergency Repair Disk after using Disk Administrator to configure volume sets, stripe sets, mirror sets, or stripe sets with parity, it might be difficult or impossible to recover data on these volumes. When you create any of these volumes, Disk Administrator updates the DISK subkey in the Registry and sets the fault-tolerant bit on the System ID field of the Partition Table for each partition or logical drive in the volume.
- You should make a copy of your current Emergency Repair Disk.
- If you convert your boot partition from the FAT file system to the NTFS file system on an x86-based computer, be sure to update your Emergency Repair Disk. The NTFS Partition Boot Sector must be on the Emergency Repair Disk, not the FAT Partition Boot Sector.
- The Emergency Repair Disk is not a replacement for backups.

For more information about creating an Emergency Repair Disk, see Help for the Repair Disk program. For more information about the *%systemroot%*\Repair and *%systemroot%*\System32\Config folders, and the Emergency Repair Disk, search the Knowledge Base. For information about the Knowledge Base, see Chapter 8, "General Troubleshooting."

Chapter 23, "Overview of the Windows NT Registry," in the *Windows NT Workstation Resource Guide*, contains more information about these keys and the *%systemroot%*\System32\Config folder.

Backing up the Master Boot Record and Partition Boot Sector

No operating system can protect itself from damage to these disk areas, which can be caused by viruses, faulty SCSI configurations, device driver problems, or power outages. You should back up the Master Boot Record on a disk every time you change partition information for primary partitions or an extended partition. You should back up a Partition Boot Sector when you format a volume, install Windows NT in the volume, or convert a volume from the FAT file system to the NTFS file system.

You can back up these disk sectors by using the Windows NT-based program, DiskProbe, or the MS-DOS-based program, DiskSave. Both of these utilities are on the *Windows NT Server Resource Kit* CD. The procedures for using DiskProbe and DiskSave are described in Chapter 7, "Disk, File System, and Backup Utilities."

If you have more than one hard disk, or more than one partition on a disk, you should back up every Master Boot Record and Partition Boot Sector. The Master Boot Record for the startup disk and the Partition Boot Sector for your system partition are the most critical. The ones for the other disks and volumes are not as critical to the startup process, but you might not be able to access files if the Master Boot Record for the disk or the Partition Boot Sector for the volume is not correct.

Chapter 6, "Troubleshooting Startup and Disk Problems," describes how to determine if you have a problem with one of these sectors.

Backing Up Data on Your Hard Disks

Regularly backing up data on hard disks minimizes data loss or damage caused by disk failures, power outages, virus infection, and many other possible computer problems. If your backup operation is based on careful planning and reliable equipment, file recovery should be relatively easy.

Windows NT includes a backup program, Ntbackup.exe, that you can use to save data to cartridge tape on computers running Windows NT Server and Windows NT Workstation. Whether you use this program or another type of tape backup system, you need to consider the following:

- What data to back up.
- What hardware to get.
- Where to locate your tape drive.
- What type of backup procedure to use.
- What backup schedule to follow.
- Where to store tapes.

Once you have designed a backup operation, document both your backup and restore procedures. After your backup system is in place, it is a good idea to periodically practice restoring an entire server, as well as individual files.

For more details about what you should cover in your backup plan, see the section titled "Backup Strategy," in Chapter 4, "Planning a Reliable Configuration." For more information about the Ntbackup program, and planning your backup procedure, see Chapter 6, "Backing Up and Restoring Network Files," in the Microsoft Windows NT Server *Concepts and Planning* book.

Understanding ARC Pathnames

Advanced RISC Computing (ARC) naming conventions are a standard for identifying the location of a file or a program on a device such as a hard disk or a floppy disk. You must understand the ARC pathname conventions to be able to create paths to use to start the computer from the Windows NT startup floppy disk.

On x86-based computers, you use ARC pathnames to describe the location of the boot partition for each instance of Windows NT installed on the computer. If you configure your boot partition as a mirror set, you need to include a path to the shadow boot partition in the Boot.ini file. Creating the path is described in "Creating Alternate Boot Selections for an x86-based Computer," presented later in this chapter.

On x86-based computers, the ARC path to the boot partition has one of the following forms:

- multi(W)disk(X)rdisk(Y)partition(Z)\*%systemroot%*
- scsi(W)disk(X)rdisk(Y)partition(Z)\*%systemroot%*

On RISC-based computers, you use ARC pathnames in the NVRAM to describe the location of:

- The folder containing Osloader.exe and Hal.dll.
- The full path to Osloader.exe.
- The path to the boot partition.
- Devices detected by the firmware, such as disks, keyboard, and video.

When you create a Windows NT startup floppy disk to use on a RISC-based computer, you need to modify the NVRAM to include the ARC path to the floppy disk. If you configure your boot partition as a mirror set, you also need to include an ARC path to the shadow boot partition in your NVRAM. Creating boot selections for both the Windows NT startup floppy disk and the shadow boot partition is described in the section titled "Creating Alternate Boot Selections for a RISC-based Computer," presented later in this chapter.

On a RISC-based computer, only the scsi() syntax is used for hard disks. The boot selection has the following parts:

- LOADIDENTIFIER is the name that appears on the **Boot** menu for this boot selection.
- SYSTEMPARTITION is the FAT partition that contains Osloader.exe and Hal.dll.
- OSLOADER is the path to the OSLOADER file itself.
- OSLOADPARTITION is the boot partition that contains the Windows NT system files.
- OSLOADFILENAME is the name of the Windows NT folder in the boot partition, with no drive letter.
- OSLOADOPTIONS indicates any debugging options that should be set up when loading Windows NT.

This is an example of a boot selection for a RISC-based computer:

- LOADIDENTIFIER=Windows NT Server 4.0
- SYSTEMPARTITION=scsi(0)disk(0)rdisk(0)partition(1)
- OSLOADER=scsi(0)disk(0)rdisk(0)partition(1)\Os\Nt40\Osloader.exe
- OSLOADPARTITION=scsi(0)disk(0)rdisk(0)partition(2)
- OSLOADFILENAME=\Winnt
- OSLOADOPTIONS=nodebug

multi(W)disk(X)rdisk(Y)partition(Z)

This form of the ARC pathname is only used on x86-based computers. On Windows NT version 3.1, it was only valid for IDE, EIDE, and ESDI disks. In Windows NT 3.5 and later, it is valid for SCSI disks as well.

The multi() syntax indicates to Windows NT that it should rely on the system BIOS to load system files. This means that NTLDR, the boot loader for x86-based computers, will be using interrupt (INT) 13 BIOS calls to find and load Ntoskrnl.exe and any other files it needs to get the system running.

Note The term multi() syntax refers to the multi(W)disk(X)rdisk(Y)partition(Z) form of the ARC pathname.

The W, X, Y, and Z parameters have the following meaning:

- W is the ordinal number of the controller, and should always be 0.
- X is not used for multi(), and is always 0.
- Y is the ordinal for the disk on the controller, and is always 0 or 1 for disks connected to the primary controller. The range is 0 through 3 for disks on a dual-channel EIDE controller.
- Z is the partition number. All partitions receive a number except for type 5 (Extended) and type 0 (unused) partitions. These numbers start at 1, as opposed to all other entries, which start with 0.

Theoretically, the multi() syntax could be used to start Windows NT from any disk on the computer. However, this would require that all disks be correctly identified through the standard INT 13 interface. But support for this varies from controller to controller, and most system BIOS only identify a single controller through INT 13.

For a configuration with IDE or EIDE disks, the multi() syntax works for up to four disks on the primary and secondary channels of a dual-channel controller. In a SCSI-only configuration, the multi() syntax works for the first two disks on the first SCSI controller (the controller whose BIOS loads first). When your computer has both SCSI and IDE or EIDE disks, the multi() syntax works only for the IDE or EIDE disks on the first controller.

scsi(W)disk(X)rdisk(Y)partition(Z)

The scsi() syntax is used for both RISC-based and x86-based computers, and is used in all versions of Windows NT. Using the scsi() syntax indicates that Windows NT needs to load a SCSI device driver and use that driver to access the boot partition.

Note The term scsi() syntax refers to the scsi(W)disk(X)rdisk(Y)partition(Z) form of the ARC pathname.

On an x86-based computer, the device driver is Ntbootdd.sys, which can be found in the root of the system disk (generally C:). It is a copy of the device driver for the disk controller in use. On a RISC-based computer, the driver is built into the firmware, as required by the RISC standards, so no file is required.

The W, X, Y and Z parameters have the following meaning when using the scsi() syntax:

- W is the ordinal number of the controller as identified by the Ntbootdd.sys driver.
- X is the SCSI ID of the target disk.
- Y is the SCSI logical unit number (LUN) of the disk that contains the boot partition. Y will almost always be 0.
- Z is the partition number. All partitions receive a number except for type 5 (Extended) and type 0 (unused) partitions. These numbers start at 1, as opposed to all other entries, which start with 0.

When using scsi() syntax, the value of W depends upon Ntbootdd.sys. Each SCSI device driver used in Windows NT has its own method of ordering controllers, although the controller order corresponds to the order that the BIOS on the controllers loads (if they are being loaded).

Additionally, if you have more than one SCSI controller, and they use different device drivers, you should only count the controllers that are controlled by Ntbootdd.sys when determining the value of the W parameter. For instance, if you have an Adaptec 2940 (which uses Aic78xx.sys) and an Adaptec 1542 (which uses Aha154x.sys), W will always be 0. What changes is the Ntbootdd.sys file:

- If you are loading Windows NT from a disk on the Adaptec 2940, Ntbootdd.sys will be a copy of Aic78xx.sys.
- If you are loading Windows NT from a disk on the Adaptec 1542, Ntbootdd.sys will be a copy of Aha154x.sys.

Example ARC Pathnames

This section contains examples that you can use as models. The first two examples are for ARC pathnames on x86-based computers. The last example is a boot selection on a DEC Alpha AXP 150 computer, and should be good for all RISC-based computers with similar disk configurations.

More Than One SCSI Controller on an x86-based Computer

This example for an x86-based computer has the following disks and controllers:

- Two Adaptec 2940s, each with two 1 GB hard disks at ID 0 and ID 1.
- One Adaptec 1542, with two 1 GB hard disks at ID 0 and ID 4.

Each hard disk has a single, 1 GB primary partition. Partitions 1 and 2 are on the first 2940, partitions 3 and 4 are on the second 2940, and partitions 5 and 6 are on the 1542. These are the ARC pathnames that you would have in the Boot.ini file if each partition had a copy of Windows NT installed in a folder named Winnt:

- Partition 1: multi(0)disk(0)rdisk(0)partition(1)\Winnt
- Partition 2: multi(0)disk(0)rdisk(1)partition(1)\Winnt
- Partition 3: scsi(1)disk(0)rdisk(0)partition(1)\Winnt
- Partition 4: scsi(1)disk(1)rdisk(0)partition(1)\Winnt
- Partition 5: scsi(0)disk(0)rdisk(0)partition(1)\Winnt
- Partition 6: scsi(0)disk(4)rdisk(0)partition(1)\Winnt

For partitions 3 and 4, Ntbootdd.sys is a copy of Aic78xx.sys. For partitions 5 and 6, Ntbootdd.sys is a copy of Aha154x.sys. For partitions 1 and 2, you could also use the following pathnames:

- Partition 1: scsi(0)disk(0)rdisk(0)partition(1)\Winnt
- Partition 2: scsi(0)disk(1)rdisk(0)partition(1)\Winnt

provided that you had an Ntbootdd.sys file that was a copy of Aic78xx.sys. However, Windows NT Setup always use the multi() syntax for the first two SCSI disks.

Both EIDE and SCSI Controllers on an x86-based Computer

This example is for an x86-based computer with the following disks and controllers:

- A dual-channel EIDE controller with three 1 GB disks, two on the primary channel and one on the secondary channel.
- An Adaptec 2940 controller with a single 4 GB hard disk at ID 3.

The three EIDE disks each have a single 1 GB partition. Partitions 1 and 2 are on the primary channel of the EIDE controller, and partition 3 is on the secondary channel.

The SCSI disk has four 1 GB partitions. Partitions 4, 5, 6 and 7 are on the SCSI disk. Each partition would have the following ARC pathnames in the Boot.ini file, assuming that each partition has Windows NT installed in the Winnt folder:

- Partition 1: multi(0)disk(0)rdisk(0)partition(1)\Winnt
- Partition 2: multi(0)disk(0)rdisk(1)partition(1)\Winnt
- Partition 3: multi(0)disk(0)rdisk(2)partition(1)\Winnt
- Partition 4: scsi(0)disk(3)rdisk(0)partition(1)\Winnt
- Partition 5: scsi(0)disk(3)rdisk(0)partition(2)\Winnt
- Partition 6: scsi(0)disk(3)rdisk(0)partition(3)\Winnt
- Partition 7: scsi(0)disk(3)rdisk(0)partition(4)\Winnt

Loading Windows NT from partitions 4 through 7 requires that the Ntbootdd.sys file be a copy of Aic78xx.sys.

Boot Selection on a DEC Alpha AXP 150

On a RISC-based computer, all boot paths are defined in the NVRAM. When creating a new boot selection for a RISC-based computer, the firmware menus prompt you for the information to enter. There are examples for defining boot selections in the section titled "Creating Alternate Boot Selections for a RISC-based Computer," presented later in this chapter.

The following example is a boot selection on a DEC Alpha AXP 150. The computer has a single hard disk at ID 0, which has a 4 MB system partition and a 396 MB boot partition. Windows NT is installed on the boot partition in a folder named Winnt and the OSLOADER folder is also named Winnt, although it is on the system partition. The boot selection has the following values:

- LOADIDENTIFIER=Windows NT Server 4.0
- SYSTEMPARTITION=scsi(0)disk(0)rdisk(0)partition(1)
- OSLOADER=scsi(0)disk(0)rdisk(0)partition(1)\Os\Winnt\Osloader.exe
- OSLOADPARTITION=scsi(0)disk(0)rdisk(0)partition(2)
- OSLOADFILENAME =\Winnt
- OSLOADOPTIONS=nodebug

Creating Floppy Disks for Starting Your Computer

For x86-based computers and RISC-based computers, you should create a floppy disk to use to start Windows NT if you cannot start up from the hard disk. For x86-based computers, you can also create a bootable floppy disk that you can use to start MS-DOS if you cannot start Windows NT.

Creating each floppy disk requires two steps:

- Format the disk.
- Copy files to the disk.

After you create the Windows NT startup floppy disk or the MS-DOS bootable floppy disk, run a virus scan program to make sure the disk is not infected. Lock the floppy disk after you finish running the virus scan. (Most floppy disks have a tab on their back side that you slide down to lock the disk. You cannot write on a floppy disk when the tab is in the locked position.)

You should test each disk before you have to use it. You should also be familiar with using the programs that you copy to the floppy disks.

Depending on your disk configuration and whether you are using an x86-based computer or a RISC-based computer, you might also have to create alternate boot selections to your Windows NT startup floppy disk. For a description of the ARC pathname conventions that you need to use in boot selections, see "Understanding ARC Pathnames," presented earlier in this chapter.

Whenever possible, you should use an ARC pathname created by Windows NT Setup during an installation. When in doubt, install a copy of Windows NT onto the disk that you want to use for the boot partition, and then copy that ARC pathname, changing the partition number and folder name as necessary.

Windows NT Startup Floppy Disk

You must format your Windows NT startup floppy disk when you are running Windows NT. You can use My Computer to format the floppy disk, or enter **format a:** at the command prompt. Both of these utilities copy the Windows NT Partition Boot Sector, which is required to load the Windows NT boot loader, to the floppy disk. Windows NT Help describes the procedure for formatting a floppy disk by using My Computer.

You can use the Windows NT startup floppy disk to start Windows NT in the following circumstances:

- The Master Boot Record or Partition Boot Sector on the system partition has a problem.
- There are problems with the disk that contains the system partition and the boot partition is on another disk.
- You are reconfiguring hard disks and you want to be able to start Windows NT from the Windows NT startup floppy disk if you are having problems.

You can configure your system or boot partition as a mirror set. A mirror set creates an identical twin for the selected partition. Creating a mirror set is similar to making a copy of a document by using a copy machine. The original partition is like the original of the document, and the shadow partition is the copy. Unlike a copy machine, however, Windows NT continually updates the shadow partition when any changes are made to the original.

If you create a mirror set for your system or boot partition, there are more situations in which you can use the Windows NT startup floppy disk.

- The system and boot partitions are the same partition, and the partition on the original disk has failed.
- The system and boot partitions are different, and the system partition on the original disk has failed. Even though the boot partition is still good, the computer cannot access the startup files on the original system partition.
- Your computer running Windows NT Server is an x86-based computer, the system partition is different from the boot partition, and the original boot partition has failed. Even though you still have the system partition, the ARC pathname in the Boot.ini file on the system partition points to the failed disk.

Note On x86-based computers, if the Partition Table in the Master Boot Record has been corrupted, especially if the information for the system partition or boot partition is all zeroes, you might not be able to start up by using the Windows NT startup floppy disk. If you have a current backup of the Master Boot Record, try starting MS-DOS from the MS-DOS bootable floppy disk, and use the DiskSave program to replace the Master Boot Record.

The files that you need to copy to the floppy disk normally have the Read Only, System, and Hidden attributes set. If the files have either the System or Hidden attribute set, they are not visible. You need to make the files visible before you can copy them.

▷ To copy files to the Windows NT startup floppy disk

1. Using My Computer or Windows NT Explorer, on the **View** menu, click **Options**.
2. On the **View** tab, select **Show all files**. Click **OK**.
3. Click the filename of the file to be copied.
4. On the **File** menu, click **Properties**.
5. In the **Attributes** box of the **General** tab, clear the **Read Only**, **System**, and **Hidden** check boxes, and click **OK**.

   Do steps 3, 4, and 5 for each file that you want to copy.
6. Select the files, and then copy them to the floppy disk.

Files to Copy for an x86-based Computer

Copy the following files from the root folder of your system partition to the floppy disk you just formatted:

- NTLDR — Windows NT boot loader program.
- Boot.ini — describes the location of the boot partitions, specified by using Advanced RISC Computing (ARC) naming conventions.
- Ntdetect.com — used for hardware detection.
- Bootsect.dos — if you want to be able to dual-boot another operating system when using the Windows NT startup floppy disk.
- Ntbootdd.sys — required only if you are using the scsi() syntax in the Boot.ini file. This file is not present in the root folder if you are using the multi() syntax.

The Ntbootdd.sys file is a renamed copy of the SCSI device driver used on your Windows NT computer. For example, if you are using the Adaptec 1542B SCSI controller, copy Aha154x.sys to the floppy disk, and then rename it to Ntbootdd.sys. If Windows NT Setup created an Ntbootdd.sys in your root folder, just copy that file.

You can find the device driver name by

- Looking in the HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap\Scsi Registry key.
- Double-clicking the SCSI Adapters option on Control Panel. In the **Devices** tab, click **Properties**. Click the **Driver** tab to see the driver filename.

Creating Alternate Boot Selections for an x86-based Computer

To start Windows NT Server after the failure of the original system or boot partition on a mirror set, you need to have a path to the shadow boot partition in your Boot.ini file. Be sure to edit the Boot.ini file to create startup options for every contingency for your configuration.

Note If you have created a mirror set of your system partition, you need to set the Boot Indicator flag in the System ID field of the shadow partition. Disk Administrator does not set the Boot Indicator flag when it creates the shadow partition. Use the DiskProbe program to set this flag. For information about DiskProbe, see Chapter 10, “Disk, File System, and Backup Utilities.” For more information about the Boot Indicator flag, see Chapter 3, “Disk Management Basics.”

If the configuration of the partitions on your original disk and shadow disk are not the same, be sure to use the correct partition number when you create the path to the boot partition on the shadow disk. For example, you might have an EISA configuration partition as the first partition on your original disk, but not on the shadow disk. If you create a mirror set for each of the other partitions on the original disk, the partition numbers will be different on each disk.

The scsi() syntax is often required to start from the Windows NT startup floppy disk, even though the multi() syntax works to start from the hard disk. If your system fails to start from the Windows NT startup floppy disk, try using the scsi() syntax instead of the multi() syntax for the pathname.

Note The system BIOS controls whether your system starts up from a floppy or hard disk. The Boot.ini file controls from which hard disk and partition the computer loads the Windows NT system files.

The Boot.ini file has the Read Only attribute set by default. Remove this attribute before editing the file. Restoring the attribute is optional. Windows NT Setup sets the attribute to prevent accidental deletion.

The rest of this section has information for creating the pathnames for three different configurations of SCSI disks and controllers. In each example, the boot partition is configured as a mirror set. There is more information presented later in this chapter about configuring the system partition as a mirror set, in the section titled "Configuring the System Partition on a Mirror Set."

For more information about ARC pathnames, scsi() and multi() syntax, and examples with IDE and EIDE disks, see "Understanding ARC Pathnames," presented earlier in this chapter.

Single SCSI Controller

To be able to start from the shadow boot partition by using the multi() syntax on computers with a single SCSI controller:

- The shadow boot partition must be on one of the first two disks on the computer.
- If the boot partition is on the second disk, the first two disks must be on the same controller.
- The BIOS must be enabled on the controller.
- You must have used the same apparent geometry when you low-level formatted each disk.
- The disk with the shadow boot partition must be the next available SCSI ID on the bus.
- The disk with the original boot partition must be powered down or removed from the bus.

The following example uses the multi() syntax.

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\FT_TEST
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\FT_TEST="Windows NT Server"
multi(0)disk(0)rdisk(0)partition(1)\FT_TEST="Windows NT Server [VGA]"
/basevideo
```

If the original disk cannot be powered down, you need to have an entry in the Boot.ini file to force NTLDR to load from the shadow boot partition. This Boot.ini file loads the system from the shadow disk of a mirror set while the original disk is still functional. Rdisk(1) refers to the ordinal number and not the SCSI ID of the disk. This example has an entry for the shadow partition.

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\FT_TEST

[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\FT_TEST="Original Disk"
multi(0)disk(0)rdisk(1)partition(1)\FT_TEST="Shadow Disk"
```

More Than One SCSI Controller

If you have more than one SCSI controller on the computer, startup after a failure to a mirror set is much easier if you use the following guidelines:

- Use identical SCSI controllers. This minimizes problems with device driver, performance, and feature set differences.
- Disable translation on the SCSI controllers.
- When using duplexed mirror sets, load balance the controllers. Include the same number of disks on each controller, if possible. This step helps with performance.
- Use the scsi() syntax in the Boot.ini file.

Here is a typical Boot.ini file for starting a computer with a mirrored boot partition, and each disk of the mirror set is on a different SCSI controller. Both the original and shadow disks are assigned SCSI ID 0 on their respective controllers.

```
[boot loader]
timeout=30
default=scsi(0)disk(0)rdisk(0)partition(1)\FT_TEST
[operating systems]
scsi(0)disk(0)rdisk(0)partition(1)\FT_TEST="Original Disk"
scsi(1)disk(0)rdisk(0)partition(1)\FT_TEST="Shadow Disk"
```

If the shadow disk was assigned SCSI ID 3, then the second entry is:

```
scsi(1)disk(3)rdisk(0)partition(1)\FT_TEST="Shadow Disk"
```

Dual-Channel SCSI Controllers

Many SCSI controllers have more than one channel on the adapter card. Even though there is a single card, it appears to the computer as if there are two separate controllers. There is quite a bit of variation in the way different manufacturers handle multiple channels. Be sure to test starting up from each of the channels and disks.

If a valid ARC path cannot be found to start the computer from the shadow boot partition, you need to make changes in the system configuration. It might be necessary to consult with the controller vendor to determine the correct configuration to be able to start Windows NT Server from the shadow partition if the original disk or SCSI channel fails. Many dual-channel controllers have an option in their Setup program, or the EISA configuration program, that enables you to specify the channel to use for the startup disk.

Files to Copy for a RISC-based Computer

When you install Windows NT on a RISC-based computer, it creates a folder, such as \Os\Winnt40, that contains the Osloader.exe and Hal.dll files. On Alpha AXP-based computers, this folder also contains several files with the .pal extension. Some or all of these files might have the system, hidden, or read-only attributes set.

A Windows NT startup floppy disk for a RISC-based computer should have a folder tree identical to the RISC-based system partition. Therefore, you should create the \Os\Winnt40 folder on the floppy disk.

Copy the following files from the \Os\Winnt40 folder on your hard disk to the same folder on the floppy disk:

- Osloader.exe
- Hal.dll
- *.pal (AXP-based computers only)

Creating Alternate Boot Selections for a RISC-based Computer

RISC-based computers start from the system firmware. You should define a boot selection in the NVRAM that points to a Windows NT startup floppy disk. You only need to use the Windows NT startup floppy disk if your original disk fails and it contains the system partition (the partition with OSLOADER). If you create a mirror set for your boot partition, you should also create a path to the shadow boot partition.

If the configuration of the partitions on your original disk and shadow disk are not the same, be sure to use the correct partition number when you create the path to the boot partition on the shadow disk. For example, if you have an EISA configuration partition as the first partition on your original disk, but not on the shadow disk, the partition numbers on your shadow disk will be different than the partition numbers on your original disk.

There are three types of RISC-based computers: Alpha, MIPS, and Power PC. The examples in the remainder of this section are for an Alpha AXP computer. The firmware menus that you use to create and change boot selections can be different for MIPS and Power PC computers. For more information about RISC-based menus, see Chapter 19, "What Happens When You Start Your Computer," in the *Windows NT Workstation Resource Guide.*

Creating a Path to the Shadow Boot Partition

To create a path to the shadow boot partition, create an alternate boot selection by using the **Add a boot selection** menu. These are the steps to get to that menu:

1. Start the computer. You see the **Boot** menu.
2. On the **Boot** menu, click **Supplementary menu.**
3. On the **Supplementary** menu, click **Setup the system.**
4. On the **Setup** menu, click **Manage boot selection menu.**
5. On the **Boot selections** menu, click **Add a boot selection.**

The following example shows the screens you would see and the selections you would make to create a path to the shadow boot partition. In this example, the system partition and the boot partition are separate partitions on the same physical disk. Both partitions are mirrored. If they were the same partition, you would have fewer entries to make to create the boot selection. If the boot partition and system partition are the same partition, that partition is considered to be the system partition.

Note The right justified arrows (<----) in these examples indicate which selection was made, or the information that was entered.

```
Thursday, 03-07-96   11:18:43 AM

Select a system partition for this boot selection:
        SCSI Bus 0 Hard Disk 0 Partition 1
        New system partition  <------------------------------------------

Enter location of system partition for this boot selection:
        Select Media:
    SCSI Hard Disk  <----------------------------------------------------
    Floppy Disk
    CD-ROM

        Enter SCSI bus number: 0  <--------------------------------------
        Enter SCSI ID: 2          <--------------------------------------
        Enter Partition: 1        <--------------------------------------

Enter the osloader directory and name: \os\winnt40\osloader.exe <-------

Is the operating system in the same partition as the osloader:
        Yes
        No <-------------------------------------------------------------

Enter the location of os partition:
        Select Media:
    SCSI Hard Disk  <----------------------------------------------------
    Floppy Disk
    CD-ROM

        Enter SCSI bus number: 0  <--------------------------------------
        Enter SCSI ID: 2          <--------------------------------------
        Enter Partition: 2        <--------------------------------------

Enter the operating system root directory: \winnt40     <---------------
Enter a name for this boot selection: Boot Shadow Disk  <---------------

Do you want to initialize the debugger at boot time:
        Yes
        No  <------------------------------------------------------------
```

After entering the data, you will be back at the **Boot selections** menu. Select **Setup menu.** On the **Setup** menu, select **Supplementary menu, and save changes.**

Creating a Path to the Windows NT Startup Floppy Disk

Once you have created the Windows NT startup floppy disk, you need to create the path to it. If you do not create a path to the floppy disk ahead of time, and your original disk fails, you will have to create a path to the Windows NT startup floppy disk to be able to start the computer.

Listed below is a sample of the configuration options used to build an alternate boot selection when the Windows NT startup files are located on the Windows NT startup floppy disk. As in the earlier examples, the arrows indicate which selections were made or which data were entered.

```
                                    Thursday, 03-07-96   11:25:13 AM

Select a system partition for this boot selection:
        SCSI Bus 0 Hard Disk 0 Partition 1
        New system partition  <-----------------------------------------

Enter location of system partition for this boot selection:
        Select Media:
    SCSI Hard Disk
    Floppy Disk  <------------------------------------------------------
    CD-ROM

        Enter floppy drive number: 0  <---------------------------------

Enter the osloader directory and name: \os\winnt40\osloader.exe <-------

Is the operating system in the same partition as the osloader:
        Yes
        No <------------------------------------------------------------

Enter the location of os partition:
        Select Media:
    SCSI Hard Disk  <---------------------------------------------------
    Floppy Disk
    CD-ROM

        Enter SCSI bus number: 0  <-------------------------------------
        Enter SCSI ID: 2          <-------------------------------------
        Enter Partition: 2        <-------------------------------------

Enter the operating system root directory: \winnt40          <---------
Enter a name for this boot selection: start from floppy disk <---------
Do you want to initialize the debugger at boot time:
        Yes
        No  <-----------------------------------------------------------
```

After entering the data, you will be back at the **Boot selections** menu. Select **Setup menu**. On the **Setup** menu, select **Supplementary menu, and save changes.**

Using Alternate Boot Selections

When you have created alternate boot selections, you start an alternate one by selecting **Boot an alternate operating system** from the **Boot** menu. This example shows selecting the alternate operating system option.

```
ARC Multiboot Alpha AXP Version 3.5-11
Copyright (c) 1993 Microsoft Corporation
Copyright (c) 1993 Digital Equipment Corporation

Boot Menu:

    Boot Windows NT Server Version 4.0
    Boot an alternate operating system <--------------------------------
    Run a program
    Supplementary menu. . .

Use the arrow keys to select, then press Enter.
Seconds until auto-boot. Select another option to override: 9
```

If you have added boot selections for the shadow boot partition and the Windows NT startup floppy disk, as described earlier in this section, and select **Boot an alternate operating system**, you would see a screen like the following.

```
ARC Multiboot Alpha AXP Version 3.5-11
Copyright (c) 1993 Microsoft Corporation
Copyright (c) 1993 Digital Equipment Corporation

Boot Menu:

    start from floppy disk
    Boot Shadow Disk
    Boot Windows NT Server 4.0 (Default)

Use the arrow keys to select, then press Enter.
```

Each time you create a new boot selection, it becomes the default selection. The preceding example shows the menu you would see if you:

- Installed Windows NT Server 4.0.
- Added the selection for the shadow boot partition.
- Added the selection for the Windows NT startup floppy disk.

You should use the **Rearrange boot selections** menu to change the default boot selection back to the hard disk. See the section titled “Using a RISC-based Computer’s Boot Menu,” in Chapter 19 of the *Windows NT Workstation Resource Guide*, “What Happens When You Start Your Computer,” for information about managing boot selections.

MS-DOS Bootable Floppy Disk (x86-based Computers)

An MS-DOS bootable floppy disk is most useful if you have FAT partitions on your startup disk (the disk that contains the system partition that you use to start Windows NT).

Your MS-DOS bootable floppy disk contains the files that you need to start MS-DOS from the floppy disk. It also should contain utilities for examining the contents of disks, copying files, and creating and formatting partitions.

Note You must start the MS-DOS operating system to format the floppy disk that you will be using for your MS-DOS bootable floppy disk. Running **format** from the command prompt does not copy the correct Partition Boot Sector to the floppy disk.

▷ **To create an MS-DOS bootable floppy disk**

1. Start MS-DOS. Put a blank floppy disk in drive A. Type **format a: /s** and press ENTER. You must specify the **/s** switch to make the floppy disk bootable. This switch causes the format program to copy the file Command.com to the floppy disk.
2. Copy other MS-DOS-based utilities that you might want to use to the floppy disk. At a minimum, you should copy these files:
   - Attrib
   - Copy
   - Format
   - Fdisk
   - Mem
   - Sys.com
   - a text editor
   - DiskSave

Training and Testing

The skill and experience of the server administrators and operators is a big factor in getting a failed computer back on line with minimal cost and disruption to your company's business. You must have support personnel who are trained in troubleshooting problems and determining what recovery procedures to use when they have found the cause of a problem.

A high level of skill can only be maintained through practice.

In preparing a training plan for server operators, you should start by imagining some typical scenarios. For example, does each operator know the steps to take to restart the computer when the original disk of a mirror set fails and the disk contains the operating system? Your plan should cover the following items:

- Does the server operator have a Windows NT startup floppy disk? On a RISC-based computer, is there an alternate boot selection available for it?
- When was the last time the Windows NT startup floppy disk was tested?
- If the problem is a controller failure, how long will it take to replace the controller? Is the hardware configuration information immediately available?

Effective training must start with the basics. Server administrators and operators should have a good understanding of Windows NT Server. The Microsoft Certified Professional Program is a good starting point.

Understanding operating system fundamentals is only the beginning. Fast, efficient recovery from crashes of a computer running Windows NT Server can only be achieved through practice, which can be done by scheduling drills that simulate a crashed computer or a disk failure. These drills should be done several times a year.

Computers that have recently been taken out of service, or are being prepared for production service, can be used for training. You can also buy and configure computers to use specifically for testing and training. Training sessions and drills are a good time to update recovery procedures, which you should document.

You should also test your disaster plan. See Chapter 4, "Planning a Reliable Configuration," for information about disaster planning.

Documenting Recovery Procedures

You should develop step-by-step procedures for getting a computer back online after various failures. You can use these procedures for:

- Testing a new computer before putting the computer into a production environment.
- Training new server administrators and operators.
- Creating a Server Operations Handbook, which should include procedures for setting up new user accounts, doing backups, maintaining the Emergency Repair Disk, and other common administrative tasks.

You should review your documentation when you make configuration changes to your computers or network. Updating the documentation is especially important when you install new versions of the operating system or change the utilities that you use to maintain your system.

Testing

Now that you have created floppy disks to use to start the computer, created an Emergency Repair Disk, and backed up the Master Boot Record and Partition Boot Sector, you should use the utilities and floppy disks to practice recovering from problems. Going through the recovery process will force you to be diligent about making backups (Emergency Repair Disk, Master Boot Record, Partition Boot Sector, and data). You will also have an idea as to how long doing each procedure should take.

You should have a computer that you can use to do the following:

- Look at the Master Boot Records, Partition Tables, and Partition Boot Sectors.
- Find the backup Partition Boot Sector on an NTFS partition.
- Deliberately destroy and recover Master Boot Records and Partition Boot Sectors.
- Delete Windows NT system files and restore them by using the Emergency Repair Disk.

If you have multiple disks in your configuration or have more than one installation of Windows NT, you should thoroughly test using your Windows NT startup floppy disk to start the computer from each boot partition. This is especially critical on x86-based computers when you have a configuration that has both SCSI and IDE or EIDI disks. Sometimes, when the multi() syntax works to start from a SCSI hard disk, you must modify the Boot.ini file on the Windows NT startup floppy disk to use the scsi() syntax.

Be sure to test recovery procedures before bringing a new computer or server into production. Every server operator should have both primary and refresher training in recovering from the most common causes of unexpected server downtime. Testing should include:

- Making sure that your Windows NT startup floppy disks do their job.
- Rebuilding mirror sets and stripe sets with parity.
- Testing UPS on the computer running Windows NT Server and on hubs, routers, and other network components.
- Testing the disaster plan.
- Restoring from your backups.

If you configure your boot partition as a mirror set, be sure to test that the path to the shadow partition on your Windows NT startup floppy disk is correct.

The following test is sufficient to determine if the ARC path is correct. The Stop error indicates that the Kernel successfully loaded from the ARC path specified. If the ARC path is wrong, the computer cannot load the Kernel and you would get a message that says so.

▷ Testing the Windows NT startup floppy disk

1. Shutdown Windows NT. Power down disks and controllers, if necessary, to test the path to the shadow boot partition.
2. Use the Windows NT startup floppy disk to start the computer.
3. If your boot selection correctly specifies the alternate ARC path to the shadow boot partition, your computer should get part way through the startup sequence, and then fail with the following Kernel STOP error:

   ```
   *** STOP: 0x0000001E (0xc0000006,0x801Abe58,0x00000000,0x00000000)
   KMODE_EXCEPTION_NOT_HANDLED
   ```

If the computer has software fault-tolerant volumes (a mirror set or a stripe set with parity), you should test the failure and replacement of one of the disks. Even though fault-tolerant volumes continue to work when one disk has failed, there is no fault tolerance until you install a replacement disk. A second disk failure during this interval will result in loss of data, because you lost the redundancy when the first failure occurred. If a backup disk is not available on site, you should know the business cost of the resulting downtime.

If you have hardware RAID arrays, your vendor's documentation should describe how to recover from disk or controller failures. Be sure to test that their procedures work for your installation.

If a network card or other network component fails on the primary domain controller (PDC), the server operator should be familiar with the procedure for promoting a backup domain controller (BDC), and demoting the failed server. Someone who is familiar with the procedure for reinstalling and reconfiguring the network card should also be available.

If a data volume fails, the server operator must be able to restore the data from the backup tape quickly and efficiently. The restore procedure should be tested frequently, both to insure the skill of the operator, and to test the quality of the backup tapes. The only way to test the quality of backup tapes is to do a full restore, which guarantees that the data are up-to-date and of consistent quality.

If your backup procedures involve the use of other computers running Windows NT Server, you must also verify that those backup and restore procedures work as expected. For information about what and how to backup, see "Backup Strategy" in Chapter 4, "Planning a Reliable Configuration."

Restoring Disk Information

If you have backed up the Registry, Master Boot Records, and Partition Boot Sectors, as recommended earlier in this chapter, you can recover from many disk and startup problems without having to reinstall Windows NT. This section describes using your floppy disks and tools to restart your computer and restore information.

One important thing that you can learn from testing is the best recovery procedure to use in a particular situation. For example, there are several methods that you can use to replace a Partition Boot Sector. You also want to learn when to use the Windows NT startup floppy disk to restart your computer, and when you have to use the repair process and the Emergency Repair Disk to replace files in order to start.

Starting Up by Using the Windows NT Startup Floppy Disk

Using the Windows NT startup floppy disk enables you to start up Windows NT so that you can troubleshoot the problem, recover from the problem, or just get the computer running again. Being able to restart the computer also enables you to finish the work that you need to get done now, and troubleshoot the problem later.

In a way, the Windows NT startup floppy disk is a troubleshooting tool. If you cannot start the computer by using the hard disk, and you can start by using the Windows NT startup floppy disk, the problem is probably one of the following:

- There are problems with the disk that contains the system partition, and the boot partition is on another disk.
- Something is wrong with the system partition itself, the Partition Boot Sector for the system partition, or the files on the system partition that are needed to start Windows NT.
- On an x86-based computer, you are reconfiguring your disks, and your first hard disk does not have a system partition yet.

Using the Emergency Repair Disk

If your system files, Registry information, or Partition Boot Sector are corrupt, and you are unable to start Windows NT by using the Last Known Good control set, you can use the Repair process in Windows NT Setup to restore your system so you can start up.

To repair a Windows NT installation, Setup needs either the configuration information that is saved on the %systemroot%\Repair folder or the Emergency Repair Disk created when you installed the operating system (or you created later by using the Repair Disk program).

You cannot repair all disk problems by using the Emergency Repair Disk. Your Emergency Repair Disk needs to match the version of Windows NT that you have installed on your computer. It also needs to have current configuration information. If you have more than one installation of Windows NT on your computer, you should have an Emergency Repair Disk for each installation. You should never use an Emergency Repair Disk from another computer.

▷ **To use the Emergency Repair Disk**

1. Start the computer from the Windows NT Setup disk 1. Insert disk 2 when it is requested. When prompted, select the option to Repair by pressing the R key. Setup displays the following options:

   ```
   [X]  Inspect Registry files
   [X]  Inspect startup environment
   [X]  Verify Windows NT system files
   [X]  Inspect boot sector
        Continue (perform selected tasks)
   ```

   A description of each of the options follows this procedure.
2. Clear all selections that you do not want to use. Click **Continue (perform selected tasks)**.
3. After disk 3, you will be prompted for the Emergency Repair Disk. Follow the instructions.

   Setup displays the following messages after it has finished the repair process:

   ```
   Setup has completed repairs.
   If there is a floppy disk inserted in drive A:, remove it.
   Press ENTER to restart your computer.
   ```

Inspect Registry Files

The Repair process in Windows NT Setup is one of several ways that you can repair Registry keys. When you select **Inspect Registry files**, Setup provides you with a list of the Registry files that it can restore. It also warns you that restoring a Registry file can result in information being lost. These are the options:

```
[ ] SYSTEM (System Configuration)
[ ] SOFTWARE (Software Information)
[ ] DEFAULT (Default User Profile)
[ ] NTUSER.DAT (New User Profile)
[ ] SECURITY (Security Policy) and
    SAM (User Accounts Database)
    Continue (perform selected tasks)
```

Select the key(s) that you want to restore by entering an X between the brackets, such as:

```
[X] SYSTEM (System Information)
```

See "Creating an Emergency Repair Disk," presented earlier in this chapter, for information about how often to create the disk, and how to save the current keys.

Inspect Startup Environment

This option verifies that the Windows NT files in the system partition are the correct ones. If any of the files that are needed to start Windows NT are missing or corrupt, Repair replaces them from the *Windows NT Server* CD.

On x86-based computers, if Windows NT is not listed in the Boot.ini file, Repair adds a Windows NT option to the file. If there is no Boot.ini file, Repair creates one.

On a RISC-based computer, Repair inspects and repairs the startup information in the NVRAM.

Verify Windows NT System Files

This option uses a checksum to verify that each file in the installation is good and matches the file that was installed from the *Windows NT Server* CD. Repair uses the Setup.log file on the Emergency Repair Disk to determine what files were installed and their checksums. The repair process also verifies that files needed to start, such as NTLDR and Ntoskrnl.exe, are present and valid. To find out if one or more service packs need to be reinstalled, check Files.lst on each service pack. The *Windows NT Server* CD is required for this option.

When Repair determines that the file on the disk does not match what was installed, it displays a message that identifies the file and asks whether you want to replace it.

Inspect Boot Sector (x86-based Computers)

This option verifies that the Partition Boot Sector on the system partition still references NTLDR, and replaces it from the Emergency Repair Disk if it does not. The Repair process can only replace the Partition Boot Sector for the system partition on the first hard disk. The Repair process can also repair the Partition Boot Sector for the system partition on the startup disk.

If you have saved the Partition Boot Sector, you can also use the procedures described in "Replacing the Master Boot Record and the Partition Boot Sector," presented later in this chapter, to replace it.

Restoring Registry Information

There are several methods that you can use to restore Registry information.

- You can use the Repair process in Windows NT Setup, which requires the Emergency Repair Disk.
- If you have used the Ntbackup program or the *Windows NT Server Resource Kit* program Regback.exe to backup the Registry, you can restore the keys from the backup media.
- You can restore just the SYSTEM key or the DISK subkey by using the procedures described in this section.

Note Windows NT Setup and the Repair Disk program store Registry information on the Emergency Repair Disk in compressed format. If you decide to restore the SYSTEM key from the Emergency Repair Disk by using either Disk Administrator or the Windows NT Registry Editor (Regedt32.exe), you can uncompress the System._ file by using one of the expand utilities. See Chapter 7, "Disk, File System, and Backup Utilities."

Restoring the SYSTEM Key

You can use the procedures in this section to restore the SYSTEM key. Because the Emergency Repair Disk also contains this key, you would generally use these procedures only if:

- You do not have an Emergency Repair Disk.
- You cannot read the Emergency Repair Disk, or the information isn't current.
- You had to reinstall Windows NT and want to have the most recent configuration information.

To use this procedure, you need to have saved the information, following the procedures in the section titled "Saving the SYSTEM Key," presented earlier in this chapter.

▷ **To use Disk Administrator to restore the SYSTEM key**

1. On the **Partition** menu, click **Configuration.**
2. On the **Configuration** menu, click **Restore.**

   A message warns you that this operation will overwrite your current disk configuration information with what was previously saved on the floppy disk. Also, any changes made during this session will be lost.
3. Insert the floppy disk containing the saved configuration information.
4. Click **OK.**

   Disk Administrator initiates a restart of your computer.

If you do not have a floppy disk that contains the current SYSTEM key, you can search for other Windows NT installations and restore the key from *%systemroot%*\System32\Config\System. In this case, the information might not match the configuration that you had been using.

▷ **To use Disk Administrator to search for and restore the SYSTEM key**

1. On the **Partition** menu, click **Configuration.**
2. On the **Configuration** menu, click **Search.**

   A message warns you that this operation will overwrite your current disk configuration information with the information from a different installation of Windows NT. Also, any changes made during this session will be lost.
3. Click **OK.**

   Disk Administrator scans your disk for other Windows NT installations, and displays a list of the installations.
4. Select an installation.
5. Click **OK.**

   Disk Administrator initiates a restart of your computer.

See Chapter 8, "General Troubleshooting," for details about troubleshooting problems by using the SYSTEM key.

Restoring the DISK Subkey

Both the Repair Disk program and Disk Administrator save the DISK subkey with the other information that they save. You would use this procedure only if you do not have current information on either of these media or you cannot read them. You might also want to use this procedure in special situations to restore just the DISK subkey, such as having to move a stripe set from one computer to another when the first computer has failed.

If you have previously saved the DISK subkey by using the Regedt32.exe (see the section titled "Backing Up the Registy," earlier in this chapter), then you can restore it.

▷ **To restore the DISK subkey by using Regedt32.exe**

1. Run Regedt32.exe.
2. Select the Registry key HKEY_LOCAL_MACHINE\SYSTEM\DISK.
3. On the **Registry** menu, click **Restore**.
4. Enter the filename or click the path to the file.

   If you are restoring the key to a remote computer's Registry, the C drive designation that appears in the scroll list refers to the C drive on the remote computer.
5. Click **OK**.

Replacing the Master Boot Record and the Partition Boot Sector

For information about determining if you have a problem with either of these disk sectors, and procedures for displaying them, see Chapter 6, "Troubleshooting Startup and Disk Problems." For information about the utilities that you can use to save, replace, and edit these disk sectors, see Chapter 7, "Disk, File System, and Backup Utilities."

Caution Be careful when you use the low-level disk editors described in this section. If you replace the wrong sector, or change individual bytes to the wrong data, you can make your problems far worse. You could destroy information that is needed by Windows NT, and your only option would be to reformat the partition or the entire disk.

Replacing the Master Boot Record

There are two problems that can occur in the sector containing the Master Boot Record:

- The Master Boot Record might be corrupt.
- The Partition Tables can be damaged.

On many x86-based computers, the fastest and simplest way to replace the Master Boot Record on an x86-based computer is to use an MS-DOS bootable floppy disk with the MS-DOS-based program Fdisk on it.

Caution Do not use Fdisk to replace the Master Boot Record if you are using a:

- Third-party translation program.
- Dual-boot program that writes information in the area between the code and the Partition Table.
- Third-party partitioning program that writes information in the area between the code and the Partition Table.

If you do not know whether you are using a program like this, you should not use this method.

▷ To replace the Master Boot Record

1. Start the computer by using the MS-DOS bootable floppy disk. (You did run virus check and lock the floppy disk after you made it, didn't you?)
2. At the **A:\** prompt, type **fdisk /mbr**.

This command replaces the Master Boot Record without altering the Partition Table at the end of the sector. There is no message or response.

If there is a Windows NT disk signature in the Master Boot Record, it is overwritten by the new Master Boot Record. Overwriting the disk signature is a problem only if the disk contains partitions or logical drives that are members of mirror sets, stripe sets with parity, volume sets, or stripe sets. You can safely overwrite the disk signature when the disk has none of these volumes, because Disk Administrator writes a new disk signature the next time you run it.

Note The **fdisk /mbr** command requires MS-DOS 5.0 or later, and only works on the first hard disk on the computer.

If you have saved the Master Boot Record by using either DiskSave (x86-based computers only) or DiskProbe, you can use one of these utilities to replace the Master Boot Record. These utilities rewrite the entire sector, including the Partition Table. If you cannot start Windows NT from the hard disk, you can usually start the computer by using the Windows NT startup floppy disk if you want to use DiskProbe to replace the Master Boot Record.

If none of these methods are available to you, you can use a low-level disk editor to copy a good Master Boot Record from another disk to the startup disk (usually disk 0 on x86-based computers). Since the smallest disk write is a sector, when you copy the Master Boot Record from another disk, you are also copying the other disk's Partition Table, which is not valid for the current disk. Therefore, you must first write down the partition information in the Master Boot Record that you are going to replace, starting from 0x1BE. You have to manually reenter the Partition Table information in hex format into the newly copied sector.

If you can start Windows NT by using the Windows NT startup floppy disk, you can use DiskMap to print a map of the disk. When you have copied a Master Boot Record from another computer of the same type (for example, another computer made by the same manufacturer with the same models of disk controllers), use DiskProbe to enter the information for the Partition Table. For information about DiskMap and DiskProbe, see Chapter 7, "Disk, File System, and Backup Utilities."

After you have replaced the Master Boot Record, you should check that it is now correct. If the Master Boot Record is not correct after replacing it, there is either a hardware problem, such as incorrect SCSI termination, or (on x86-based computers) there is a virus in memory that is trapping interrupt (INT) 13 calls. You have to isolate and correct the problem that is corrupting the Master Boot Record.

Repairing the Partition Table

If you have not changed the Partition Table since you saved the Master Boot Record, replacing the Master Boot Record also replaces the Partition Table.

If you have made only minor changes to the Partition Table since you saved the Master Boot Record, you might be able to use the information on the backup to help you repair the Partition Table. For instance, if the only change you have made is to convert a partition from FAT to NTFS, just change the System ID byte from 06 to 07.

If your disk has more than one partition, and you did not save the Master Boot Record or write down the information, you might be able to reconstruct it. You can use DiskProbe to edit the Partition Table.

Note If you will have to completely rebuild your Partition Table, either because of corruption or you had to copy the Master Boot Record from another computer, it might be safer to completely rebuild your disk. In this case, you should back up all of your files to tape (or another computer), recreate and reformat the partitions on the hard disk, and restore the data. If the disk with the invalid Partition Table contains your system or your boot partition, you will probably have to use the Windows NT startup floppy disk to start your computer.

Replacing the Partition Boot Sector

There are several methods that you can use to replace the Partition Boot Sector.

If you have saved the Partition Boot Sector by using either the DiskProbe or the DiskSave program, you can replace the sector from the backup. This is the fastest way to replace the Partition Boot Sector.

For NTFS volumes, there is one other alternative. When you create or reformat an existing volume as an NTFS volume, the NTFS file system writes a duplicate of the Partition Boot Sector at the end of the volume (Windows NT version 4.0) or at the logical center of the volume (Windows NT 3.51 and earlier). You can use the Windows NT-based program DiskProbe to locate and copy this sector to the beginning of the volume. There are also third-party MS-DOS-based disk utilities that you can use to locate and copy this sector.

You might want to display the Partition Boot Sector to make sure it is now correct. Once you have copied the Partition Boot Sector to the first sector of the partition, you should be able to start Windows NT.

Restoring Windows NT Files on the System Partition

If any of the files that Windows NT installs on the system partition are missing or corrupt, you cannot start your computer. These are the files:

x86-based file	Folder	RISC-based file	Folder
NTLDR	root	Osloader.exe	\Os\Winnt40
Boot.ini	root	no equivalent	
Bootsect.DOS	root	no equivalent	
Ntdetect.com	root	no equivalent	
Ntbootdd.sys (SCSI only)	root	no equivalent	
Hal.dll is in the boot partition		Hal.dll	\Os\Winnt40
no equivalent		*.pal files (Alpha-based computers)	\Os\Winnt40

You can restore any of these files by using the Repair process and the Emergency Repair Disk. However, because you copied these files to your Windows NT startup floppy disk when you created it, you can use that disk to restore them.

▷ **To use the Windows NT startup floppy disk to restore system files**

1. Start your computer by using the Windows NT startup floppy disk.
2. Replace individual files, or replace all of the files, by copying them from the Windows NT startup floppy disk to the corresponding location on your system partition.

Configuring and Using Mirror Sets and Stripe Sets With Parity

This section discusses using features that are available in Windows NT Server software to implement mirror sets and stripe sets with parity. It does not discuss configuring or using hardware RAID arrays.

There are four topics in this section:

- A general overview of mirror sets ands stripe sets, and the information that you see in Disk Administrator.
- Configuring the system partition on a mirror set, which describes the best way to implement a mirror of the system partition. It also describes starting up by using the shadow system partition, and rebuilding the mirror set after you have repaired or replaced the failed component.
- Procedures for creating mirror sets and stripe sets with parity.
- Guidelines and procedures for reconstructing mirror sets and stripe sets with parity after you have repaired or replaced the failed component.

Overview

A mirror set creates an identical twin for the selected partition. Creating a mirror set is similar to making a copy of a document by using a copy machine. The original partition is like the original of the document, and the shadow partition is the copy. Unlike a copy machine, however, Windows NT continually updates the shadow partition when any changes are made to the original.

Note In this chapter and other chapters in the "Reliability and Recoverability" part of this book, the terms original disk and original partition refers to the original partition, and shadow disk and shadow partition refers to the disk or partition that contains the copy.

A stripe set with parity is composed of strips of equal size on each disk in the volume. One of the strips in each stripe is used for parity information.

For a more complete description of mirror sets and stripe sets with parity, including the parity information, see the section titled "Planning a Fault-Tolerant Disk Configuration," in Chapter 4, "Planning a Reliable Configuration."

Because a mirror set or stripe set with parity requires more than one disk, you might have to buy and install additional disks. To increase the redundancy, you might want to buy more disk controllers. It is a good idea to have a spare disk and controller of the same type available. If you are using RAID hardware, follow the manufacturer's installation instructions. When using software fault tolerance, using SCSI controllers is more efficient and provides an additional recovery mechanism of sector sparing. For more information, see "Processing of Disk Errors" in Chapter 3, "Disk Management Basics."

You use Disk Administrator to create and recover mirror sets and stripe sets with parity. When you select one of the partitions of a mirror set or stripe set with parity, Disk Administrator displays information about the volume in the lower left corner of the window. The information looks like:

```
<Volume> #N [status]
```

where

- <Volume> is stripe set, mirror set, or stripe set with parity.
- N is the number of the volume.

The following table shows the possible status values.

Table 5.1 Disk Administrator status information for stripe sets, mirror sets, and stripe sets with parity

Mirror set	Stripe set with parity	Stripe set	Meaning
HEALTHY	HEALTHY	HEALTHY	The volume has been initialized, and there are no errors.
NEW	NEW	NEW	Appears immediately after the volume has been created, but before shutting down the system to begin generation of the volume.

Table 5.1 Disk Administrator status information for stripe sets, mirror sets, and stripe sets with parity ***(continued)***

Mirror set	Stripe set with parity	Stripe set	Meaning
REGENERATING	INITIALIZING	INITIALIZING	Generation of the volume has been started, but is not yet completed.
RECOVERABLE	RECOVERABLE	Not applicable	Appears when one of the partitions in the volume has been lost, but the other partition(s) is (are) undamaged. This message also appears when one partition loses synchronization with the other(s).

Configuring the System Partition on a Mirror Set

If you want to upgrade an existing Windows NT installation on a mirror set, you have to break the mirror set before starting Windows NT Setup. This means that you have to recreate the mirror set of the boot partition to have the Windows NT installation copied to the shadow partition.

The term duplexed mirror set means that each disk in the mirror set has its own disk controller. The controllers in a duplexed mirror set should be the same manufacturer and model

On x86-based computers, if your system partition is on a SCSI disk, you must have the BIOS enabled on its SCSI controller. Otherwise, the system BIOS will not be able to find Master Boot Record.

If you have created a mirror set of your system partition by using two SCSI controllers, you must make sure that you are using the same translation on the two controllers. If you are using different translations on your original disk and shadow disk, you will have to use the Windows NT startup floppy disk to start from the system partition on the shadow disk if the original disk fails. In addition, if you reconstruct the system partition on the original disk from the one on the shadow disk, you will not be able to start from either hard disk, and will always have to use the Windows NT startup floppy disk to start Windows NT.

You use the translation feature on SCSI controllers to set the translation mode. Because you have to disable the BIOS on the second controller, Windows NT always translates the geometry of the shadow partition to 32 sector per track and 64 heads per cylinder. Therefore, it is important to make sure the SCSI controller for the original disk is using the same translation. When their BIOS is enabled, most SCSI controllers default to a large disk translation scheme if the disk is larger than 1 GB. The geometry for this translation is 255 heads and 63 sectors. Most SCSI controllers have some method to manually configure the controller to use the 32 sectors, 64 heads translation. Using 32 sectors and 64 heads on the controller for the original disk insures that both disks are as close to a true sector by sector mirror image as possible. Recovery from failures of the original disk or its controller is much easier if the geometry of both disks is the same. The only disadvantage is that the system partition must be within the first 1 GB, or 1024 cylinders, of the disk.

Disabling the controller BIOS is different than turning off translation. Problems occur on x86-based computers because you need to have the BIOS enabled on the first controller in order to start the computer from the hard disk. If you do not turn off translation on both controllers, and the first controller fails, you need to enable the BIOS on the second controller. The translation on the second disk is now different than when you low-level formatted it, and you cannot use the disk. If your controller does not have the option to turn off translation, you can always leave the BIOS disabled, and use the Windows NT startup floppy disk to startup from the second controller.

Note If you disable the BIOS on both controllers, it does not matter if translation is enabled or disabled. However, you will not be able to start up from the hard disk if the BIOS is disabled.

If you get a message "drive is too small" when you try to reconstruct the mirror set after the failure of the original disk, the translation is not consistent.

Having two different SCSI controllers makes a trouble-free system restart in the event of a hardware failure far more difficult. If you use two different SCSI controllers in your mirror set, you need to have a different SCSI device driver for each controller. When you create your Windows NT startup floppy disk, you need to copy both device drivers to the disk. Then, when you use the Windows NT startup floppy disk, you need to rename the one for the controller that you will be using to Ntbootdd.sys. Alternatively, you can create two separate Windows NT startup floppy disks, copy the appropriate driver to each one, and rename it to Ntbootdd.sys.

You can use EIDE controllers for a mirror set of the system partition, and using them is less complex than using SCSI controllers on x86-based computers. EIDE disks usually require motherboard BIOS support for logical block addressing (LBA), and each motherboard usually does the LBA in the same way for all EIDE disks.

If you are going to mirror an IDE or EIDI disk with a SCSI disk, you should not mirror your system partition. You will probably have problems with incompatible geometry, and will always have to use the Windows NT startup floppy disk to restart after a failure of your original system partition. However, you can safely mirror your boot partition.

Recovering from the failure of the original system partition can be difficult if you do not plan your configuration and recovery procedure carefully. Following the procedures in this section give you the easiest and most reliable recovery process.

▷ **To create a mirror set of the system partition**

1. Enable the BIOS on the controller for the disk with the system partition, but make sure that translation is turned off if the disk is larger than 1 GB. Use the SCSI setup program to turn off translation and set the geometry to use.
2. Disable the BIOS on all other SCSI controllers.
3. Make sure that the two disks you are mirroring are identical models.
4. Low-level format both disks by using the controller that has the BIOS enabled.
5. Make sure the partition layout on the two disks is the same, and that the cluster size is the same on both partitions.
6. Install Windows NT on the original disk.
7. Create the mirror set.
8. When the mirror set has finished initializing, set the Boot Indicator flag for the shadow disk. Disk Administrator does not set this flag. You can use the DiskProbe program, described in Chapter 7, "Disk, File System, and Backup Utilities," to set this flag.

When the shadow disk or controller fails, recovery is basically the same as recovering from the failure of any mirror set. You should follow the same guidelines for having identical disks and controllers as described earlier in this chapter. You can use the procedures in "Recovering a Mirror Set," presented later in this chapter, to rebuild the mirror set when you have the replacement hardware.

When the original disk or controller fails, Windows NT automatically starts using the shadow disk for all accesses. However, sometimes a problem is not detected until you try to start the computer. In this case, you can use the Windows NT startup floppy disk to start from your shadow partition. This method minimizes the amount of time that you are down, and allows you to schedule the reconstruction of the mirror set for a less busy time. You might need to power down the failed disk or remove it from the SCSI bus before attempting to start from the shadow partition.

When you can schedule down time on the computer, use one of the following two methods to reconstruct the mirror set.

▷ **Simplest method to reconstruct the mirror set**

1. While running Windows NT Server:
   - Break the mirror set, if you did not break it earlier. You can break the mirror set at any time after the failure, but you need to break it before you can start from your shadow disk without using the Windows NT startup floppy disk. The procedure to break the mirror set is described in the section titled "Recovering a Mirror Set," presented later in this chapter.
   - If you did not set the Boot Indicator flag in the System ID field of the system partition on the shadow disk, you can set it now by using Disk Administrator, because you have broken the mirror set. See Chapter 3, "Disk Management Basics," for information about this flag.
   - Shutdown the computer.

Note If you do not break the mirror set, you might get a Stop error 0xC000021A. This error occurs when there is an inconsistency in the Registry key HKEY_LOCAL_MACHINE\SYSTEM. When you get this error after a failure of the mirrored system partition, it is usually because the FT bit is set in the Master Boot Record, but the HKEY_LOCAL_MACHINE\SYSTEM\DISK Registry subkey does not have information about the mirror set. Your only recovery in this situation is to use a third-party low-level disk editor to clear the high order two bits of the Boot Indicator field for this partition. You will still have to break the mirror set after you startup Windows NT.

2. Remove the old original disk.
3. Replace the original disk with the shadow disk, changing SCSI IDs, if necessary.

4. Restart the computer. You might get a STOP 0x7b, a missing operating system error message, or similar errors that indicate there is something wrong with the Master Boot Record or the system partition on the shadow disk. You can do the following things to try to start from the shadow disk:
   - On x86-based computers, start the computer from your MS-DOS bootable floppy disk and replace the Master Boot Record by entering **fdisk /mbr** at the MS-DOS prompt. See the caution about using this switch in the section titled "Replacing the Master Boot Record," presented earlier in this chapter.
   - To replace the Windows NT files on the system partition, use Windows NT Setup, and select repair. Select the options to **Inspect boot sector** and **Inspect startup environment**, but do not select **Inspect Registry files** or **Verify Windows NT system files.**
5. If you cannot start from the shadow disk, move the shadow disk back to its old position. Use the next procedure described in this section to reconstruct the mirror set.
6. If you can start from the shadow disk:
   - Shut down the computer.
   - Install a new disk in the shadow disk's previous location.
   - Low-level format the new shadow disk, if necessary. If you have a duplexed mirror set, and the disk is larger than 1 GB, be sure to use the same translation (32 sectors, 64 heads) on the two disks.
   - Start up the computer.
   - Use Disk Administrator to recreate the mirror set. Make sure to set the Boot Indicator flag on your newly created shadow disk by using DiskProbe.

There are times when the preceding procedure will not work, such as when you are using a Compaq computer with the EISA configuration partition as the first partition on the system disk.

The next procedure always works to recover from a failure of the original system partition if you set up your configuration as described earlier in this section, and if the shadow disk was not also damaged by the failure. You will need to schedule some down time for the computer when you are ready to replace the original disk.

The next procedure takes longer than simply swapping the original disk for the shadow disk and making the shadow disk the new original, but it guarantees that you have a stable system. After you have had one disk failure on the computer, you might not want to rely on the remaining disk as the original disk, because it might have been affected in some way. This procedure is much safer from a hardware as well as a software standpoint.

▷ Most reliable method to reconstruct the mirror set

1. While running Windows NT Server:
    - Break the mirror set, if you did not break it earlier. You can break the mirror set at any time after the failure, but you need to break it before you can start from your shadow disk without using the Windows NT startup floppy disk. The procedure to break the mirror set is described in the section titled “Recovering a Mirror Set,” presented later in this chapter.
    - If you did not set the Boot Indicator flag in the System ID field of the system partition on the shadow disk, you can set it now by using Disk Administrator, because you have broken the mirror set. See Chapter 3, “Disk Management Basics,” for information about this flag.
    - Back up the entire shadow disk, including the Registry.
    - Shutdown the computer.
2. Replace the original disk with a new disk of the same make and model. (If you had to move the shadow disk or change its SCSI ID because of the failure, change it back to the original configuration.)
3. Low-level format the replacement disk, if necessary. If you have a duplexed mirror set, and the disk is larger than 1 GB, be sure to use the same translation for the two disks (32 sectors, 64 heads).
4. Install Windows NT on the new disk, creating and logically formatting partitions, as necessary. Make sure you install Windows NT into the same folder as your previous installation was in.
5. Start the new installation of Windows NT. Restore the backup, including the Registry.
6. Restart the computer from the original disk, and verify that the restore recovered all of the information on the disk.
7. Use Disk Administrator to delete the partitions on the shadow disk, and re-create the mirror. You might need to create other partitions on the shadow disk so you have the same layout on the two disks. Make sure to set the Boot Indicator flag on your newly created shadow system partition by using the DiskProbe program, as described in Chapter 7, “Disk, File System, and Backup Utilities.”

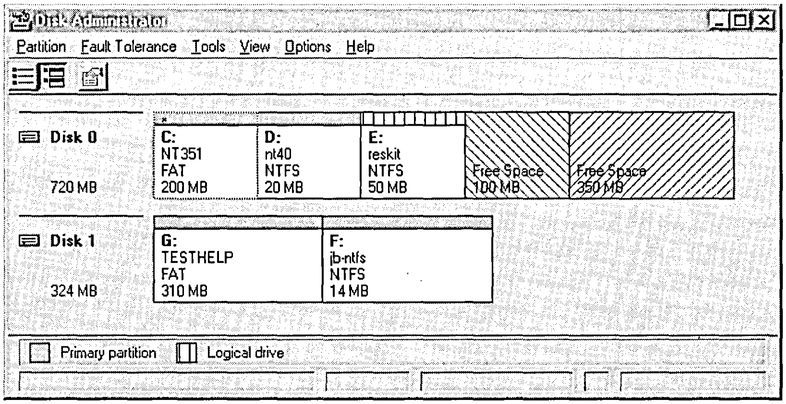
Creating a Mirror Set or Stripe Set With Parity

If you are using all SCSI disks, it is a good idea to use the same translation on all SCSI controllers whose disks are members of a mirror set or stripe set with parity. On x86-based computers, you should disable translation any time that you do not need to do the following:

- Dual-boot with MS-DOS.
- Have a system partition or boot partition larger than 1 GB.

The following Disk Administrator screen shot shows three things:

- The unpartitioned space that you can use to create a mirror set of primary partitions F and G is the 350 MB are on the right side of disk 0.
- You cannot use the free space to the right of logical drive E for a mirror set, because it is free space at the end of an extended partition.
- You cannot create a stripe set with parity, because you have only two disks. If you had a third disk, with unpartitioned space, you still could not create a stripe set with parity. You need at least three disks with unpartitioned space, and disk 1 is completely partitioned.



Note The key below the status bar in Disk Administrator identifies what kinds of volumes are configured on the computer. For example, if you created a mirror set, you would see **Mirror set** in addition to **Primary partition** and **Logical drive**.

When you create a mirror set or stripe set with parity, Disk Administrator creates a primary partition for the areas of unpartitioned space that you select on each disk. You cannot use free space on a disk for either of these volumes if:

- There are already three primary partitions, or two primary partitions and one extended partition defined on the disk.
- The free space is part of an extended partition.

Disk Administrator Help contains details about procedures for creating mirror sets and stripe sets with parity.

Creating a Mirror Set

If you need to dual-boot with MS-DOS, you cannot mirror your system partition. If you need a boot partition larger than 1 GB, you need to enable translation. However, you should not mirror your system partition if it goes beyond 1 GB.

When you want to create a mirror set, the original partition that you want to mirror must already exist. You must have an area of unpartitioned disk space at least as large as the original partition on another disk connected to the computer.

If the unpartitioned space you select for the partition on the shadow disk is larger than the existing primary partition, the space left over remains unpartitioned space, and can be used to create other partitions (if there are fewer than four partitions on the disk). For example, if you create a mirror set using the configuration shown in the preceding Disk Administrator screen shot, there will be unpartitioned space following the shadow partition. However, you cannot use that unpartitioned space, because you will have three primary partitions and one extended partition on disk 0.

▷ **To use Disk Administrator to establish a mirror set**

1. Select the partition that you want to duplicate. Press CTRL, and click an area of unpartitioned space on another disk that is the same size or larger than the partition you selected.
2. On the **Fault Tolerance** menu, click **Establish Mirror**.

   Disk Administrator creates an equal-sized partition in the unpartitioned space for the mirror set, and assigns the drive letter to the mirror set.

   Note In some cases, where a mirror set is being established, a slightly larger amount of space is required for the shadow partition than for the partition on the original disk. This happens when geometry differences exist between the disks.

   If you are mirroring the system partition, Disk Administrator displays a message box reminding you to create a Windows NT startup floppy disk.
3. On the **Partition** menu, click **Commit Changes Now**. Select **Yes** in the **Commit Changes** dialog box.

   A **Confirm change/restart** dialog box is displayed; again select **Yes**. A message box prompts you to update your Emergency Repair Disk, which you should do by using the Repair Disk program (Rdisk.exe). Finally, you are prompted to shutdown, and restart the system. The only option is **OK**. The system shuts down and restarts.

4. After the system restarts, open Disk Administrator. The status bar displays a message **Mirror Set #N [INITIALIZING]**.

   The drive letter, volume label, and size of the mirror set appear in red text. This visual cue indicates that the system is generating the partition in the background. While the text is highlighted in red, fault tolerance is not functional.

Note In the Disk Administrator display, information about the partition is not updated automatically. If you click on the partition, the information for that volume updates. You can also check the event log to know when the generation of the shadow partition is finished.

Creating a Stripe Set With Parity

When you want to create a stripe set with parity, you must have at least three unpartitioned areas on different disks connected to the computer. Even though a partition is not completely full, you cannot use the space that is part of an existing partition. If you do not have enough unpartitioned space available, but have enough unused areas in existing partition(s), you can create unpartitioned disk areas by:

- Backing up the data in the existing partition(s).
- Deleting the partition(s).
- Creating a new, smaller partition, leaving part of the disk unpartitioned.
- Restoring the data to your new partition.

As with a mirror set, if the areas of unpartitioned space that you select for a stripe set with parity are not all the same size, the space left over can also be used for other partitions.

When using a stripe set with parity, performance should be better if you put the operating system and page file on a controller other than the one used for the strip set with parity. You can use different types of controllers for the operating system and the stripe set with parity when you have them on different controllers. For more suggestions, see "Stripe Set With Parity" in Chapter 4, "Planning a Reliable Configuration."

▷ To use Disk Administrator to create a stripe set with parity

1. Select areas of unpartitioned space on three to 32 hard disks by selecting an area of unpartitioned space on the first disk, pressing CTRL, and then choosing additional areas of unpartitioned space on other hard disks.

   If the areas selected vary in size, then the smallest area is used as the base partition size. For example, if the unpartitioned space consists of a 700 MB on disk 1, and 900 MB each on disks 2 and 3, only 700 MB on each disk is used as part of the stripe set with parity.

2. On the **Fault Tolerance** menu, click **Create Stripe Set With Parity**.

   Disk Administrator displays the minimum and maximum sizes for the stripe set with parity. The default size is the maximum allowed for the disk areas selected.

   The size you choose is the total disk space that is used, not the size available for data on the stripe set with parity. For example, if four 200 MB partitions are selected:

   - Disk Administrator displays 800 MB for the maximum size.
   - You have 600 MB available for data, if you select the maximum size. The other 200 MB are used to store the parity information.
   - If you want to use less than the maximum size, the formula is

     ```
     data size = (total size) - ((total size) / (number of disks))
     ```

3. On the **Create Stripe Set With Parity** dialog box, type the size of the set that you want to create, and then click **OK**.

   Disk Administrator divides the total size that you enter by the number of disks in the set to create equal-sized, unformatted primary partitions on each of the selected disks. It then assigns a single drive letter to the collection of partitions that make up the stripe set with parity. Disk Administrator displays the stripe set with parity as New Unformatted space.

4. On the **Partition** menu, select **Commit Changes Now**.

   You are prompted to save your disk configuration changes. Select **Yes**.

5. You are notified that you need to restart the computer for the changes to the configuration to take effect. Click **Yes** in the dialog box.

6. You are notified that the disks were updated successfully.

   This means that the new configuration information has been saved in the Registry. This dialog box also prompts you to update the Emergency Repair Disk.

7. You are prompted to restart your computer so that the changes can take effect. The only option is **OK**. The system automatically closes all open applications and restarts the system.

8. When restart is completed, open Disk Administrator again. On the **Tools** menu, click **Format**. You can format a stripe set with parity with either the FAT file system or the NTFS file system. When you have entered all of the information in the **Format** dialog box, click **Start**. The stripe set with parity begins initializing immediately. Disk Administrator displays a message **Stripe Set with Parity #N [INITIALIZING]** on the status bar. The drive letter, volume label, and size of the stripe set with parity appear in red text during initialization.

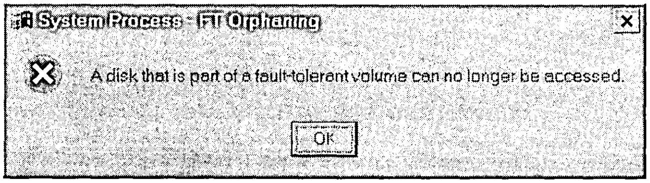
Recovering a Mirror Set or Stripe Set With Parity

The process of error detection and recovery for software fault-tolerant volumes is very similar for both mirror sets and stripe sets with parity. The Windows NT response to the problem depends on when the problem occurred. For recovery of a hardware fault-tolerant volume, see the documentation for the controller that you are using.

When a disk that is part of a mirror set or a stripe set with parity fails during normal operation, it becomes an orphan. When FtDisk (the fault-tolerant driver) determines that a disk has been orphaned, it directs all reads and writes to the other disk(s) in the set.

It is important to note that the process of orphaning a partition does not occur during a read, only during a write. The read cannot possibly affect the data on the disks, so performing orphan processing is not necessary.

The following error message is displayed:



A disk that is part of a fault-tolerant volume can no longer be accessed.

The operating system should continue to work normally. Users accessing resources over the network should not be affected.

You should back up important data immediately, since the volume is no longer fault tolerant. Use a new tape for backup, not an existing tape. You should replace the failed disk and begin the recovery of the mirror set or stripe set with parity as soon as possible.

During system initialization, if the system cannot locate a partition in a mirror set or a stripe set with parity, it logs a severe error in the event log, marks the partition as an orphan, and uses the remaining partition(s) of the mirror set or stripe set with parity. The system continues to function by using the fault-tolerant capabilities inherent in such volumes.

In Disk Administrator, if you select a mirror set or stripe set with parity that has had a failure, you see a message in the status bar that says **<Volume> #N [RECOVERABLE]** where <Volume> is either mirror set or stripe set with parity.

Recovering a Mirror Set

There are different methods to use for recovering mirror sets, depending on which partition failed, and whether it contains the system or boot partition. When the disk or controller for the original system or boot partition fails, you will probably have to use the Windows NT startup floppy disk to start from the shadow partition, or reconfigure the shadow disk as the original disk. For information about when you need to use the Windows NT startup floppy disk, see "Creating a Windows NT startup floppy disk," presented earlier in this chapter.

If the failure does not cause any disruption in service, you can continue running in a non-fault-tolerant configuration and schedule a time to regconstruct the mirror set. This activity could occur during a normally scheduled maintenance period or during a less busy time.

You must first break the mirror-set relationship to expose the remaining secondary partition as a separate volume. This step prevents problems when restarting the system.

▷ **To break the mirror set**

1. Open Disk Administrator and select the mirror set you want to break.
2. On the **Fault Tolerance** menu, click **Break Mirror**.
3. In the **Confirm** message, select **Yes**.

The remaining, working member of the mirror set receives the drive letter that was previously assigned to the complete mirror set. The orphaned partition receives the next available drive letter, or whatever letter you want to assign.

You can now shut down the system and replace the failed disk. The failed disk can be replaced with any disk that is the same size or larger. It is a good idea to use a disk as similar to the remaining disk as possible. If the failed disk contained the system partition, see the section titled "Configuring the System Partition on a Mirror Set," presented earlier in this chapter.

Note When you move or replace a disk that was at the end of a SCSI bus, be sure that you terminate only the disk that is now at the end of the bus.

▷ **To recontruct the mirror set**

1. Perform a low-level format of the new disk on the same controller that will be used with the new disk.

   This step eliminates any possibility of translation problems.

   If the failed disk was the shadow disk, use the same SCSI ID as the failed disk.

   If the failed disk was the original disk, you might want to swap SCSI IDs, so that the remaining disk becomes SCSI ID 0.
2. Restart the system.
3. Once you have restarted the computer, follow the procedure in the earlier section titled "Creating a Mirror Set or Stripe Set With Parity" to reconstruct the mirror.

   This step requires a second restart of the system to reconstruct the mirror.
4. After the mirror initialization is complete, update your system information, if necessary, as described in the section "Maintaining Configuration and Essential System Information," earlier in this chapter.

Recovering a Stripe Set With Parity

When a member of a stripe set with parity is orphaned, you can reconstruct the data for the orphaned member from the remaining members. Use the following procedure to initiate the recovery of the stripe set with parity. When you restart the computer, the FtDisk program reads the information from the strips on the other member disks, reconstructs the data of the missing member, and writes it to the new member.

If your computer running Windows NT Server has failed and you need to have the data available sooner than the expected repair time, you can move the disks containing the stripe set with parity to another computer and build the Registry key HKEY_LOCAL_MACHINE\SYSTEM\DISK by using the FtEdit program. The procedure is described in Chapter 7, "Disk, File System, and Backup Utilities."

▷ **To reconstruct a stripe set with parity**

1. Open Disk Administrator, and select the recoverable stripe set with parity.
2. Select an area of unpartitioned space of the same size or larger on the replacement disk.

   If the failure is due to a power failure or cabling failure on a single device, you can regenerate within the orphaned member of the original stripe set with parity once the hardware state is restored.
3. On the **Fault Tolerance** menu, choose the **Regenerate** command.
4. Quit Disk Administrator, and restart your computer.

The reconstruction process occurs in the background. If you open Disk Administrator, the message in the status bar is **Stripe set with parity #n [INITIALIZING].**

You might receive the following error message when attempting to reconstruct a stripe set with parity:

```
The drive cannot be locked for exclusive use...
```

This error occurs if Disk Administrator does not have exclusive access to the stripe set with parity, which happens if the page file, or some other system service, like Microsoft SQL Server or Microsoft Systems Management Server, is accessing the disk. You must temporarily shutdown these services and relocate the page file to regenerate the stripe set with parity.

Note You should not put your page file on a stripe set with parity, because it degrades performance. If you want to have your page file on a fault-tolerant volume, use a mirror set instead.

Using an Uninterruptable Power Supply

An Uninterruptable Power Supply (UPS) supplies power to a computer system in the event of a power fluctuation or complete power loss. The UPS has electronics built in that constantly monitor line voltages. If the line voltage fluctuates above or below predefined limits, or fails entirely, the UPS supplies power to the computer system from built-in batteries. The UPS converts the Direct Current (DC) battery voltage into the Alternating Current (AC) voltage required by the computer system. The changeover to batteries must take place very rapidly or the computer can lose data.

Most UPS devices are one of the following types:

- Online UPS. You connect an online UPS between the main power and the computer to constantly supply your computer system with power. Connecting it to the main power keeps its battery charged. This method provides power conditioning, which means that it removes spikes, surges, sags, and noise.
- Standby UPS. This device is configured to provide either the main power or its own power source and to switch from one to the other as necessary. When the main power is available, the UPS device connects the main power directly to the computer and monitors the main power voltage level. When the main power fails or the voltage falls below an acceptable level, the UPS device switches the power fed to the computer from the main power to its own power.

UPS systems provide a hardware interface that can be connected to the computer. Using appropriate software, this interface enables an orderly handling of the power failure, including performing a system shutdown before the UPS batteries are depleted. Without such software, an orderly shutdown of the system is not possible without human intervention.

The most important consideration in selecting a UPS product is to use only hardware that is listed on the Windows NT 4.0 Hardware Compatibility List (HCL). Other items to consider are:

- Whether to use a UPS per computer or have larger capacity, centralized UPSs that protect multiple computers.
- What type of UPS you want (standby and online are the most common).
- The sizing for the UPS (how big the UPS needs to be to protect the load).
- How long it needs to keep running before automatic shutdown.
- Which features you want to have in your UPS, such as:
  - Continuous conditioning of the incoming power to provide clean, steady power.
  - A Simple Network Management Protocol (SNMP) card that you can plug into the UPS and use network management software to remotely monitor the UPS.
  - Software that logs UPS information to the event log, or produces statistics.
  - Software for testing the integrity and reliability of the UPS battery.

Windows NT has built-in UPS functionality that takes advantage of the special features that many UPS systems provide. These features ensure the integrity of data on the system and provide for an orderly shutdown of both the computer system and the UPS should a power failure last long enough that the UPS batteries become depleted. In addition, users connected to a computer running Windows NT Server can be notified that a shutdown will occur and new users are prevented from connecting to the computer. Finally, damage to the hardware from a sudden, uncontrolled shutdown can be prevented.

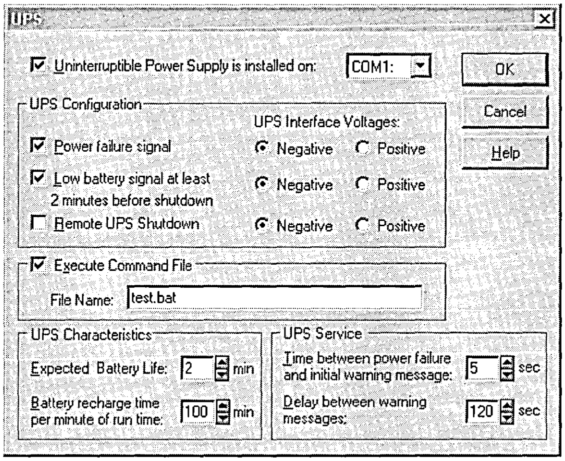
Some vendors also provide a user interface for configuring the UPS, which you can use instead of the one provided in Windows NT.

This section contains a general overview of the Windows NT UPS service and discusses configuring and using the UPS on a computer running Windows NT. To fully protect your network, you should also install UPSs on network devices such as routers, hubs, and bridges. For the best protection, install UPSs on the cables that connect your computer and modem, telephone, printer, and network equipment.

For more information about UPS, see Chapter 7, "Protecting Data," in the Windows NT Server *Concepts and Planning* book.

Configuring the UPS Service

Use the UPS option on Control Panel to configure the Windows NT UPS service. The following dialog box is displayed:



☑ Uninterruptible Power Supply is installed on: COM1:

UPS Configuration

UPS Interface Voltages:

☑ Power failure signal ● Negative ○ Positive

☑ Low battery signal at least 2 minutes before shutdown ● Negative ○ Positive

☐ Remote UPS Shutdown ● Negative ○ Positive

☑ Execute Command File

File Name: test.bat

UPS Characteristics

Expected Battery Life: 2 min

Battery recharge time per minute of run time: 100 min

UPS Service

Time between power failure and initial warning message: 5 sec

Delay between warning messages: 120 sec

You need to specify the serial port that the UPS is installed on. The information that you enter in the other fields depends upon your specific UPS and how you want to use it.

When selecting the signals to use in the **UPS Configuration** group box, the interface voltages you specify are those that indicate an active state for the signal. In other words, if you select a negative interface voltage for the power failure signal, this means that the signal is normally positive and changes to negative when a power failure occurs.

If you want the UPS service to run a command file when a power failure occurs, select the **Execute Command File** check box and enter the name of the file. Windows NT terminates the command file if it does not complete its execution within 30 seconds. The command file should not contain a dialog box, since waiting for user input could exceed the 30 second requirement.

The command file must reside in your %systemroot%\System32 folder and have one of the following extensions:

- .exe
- .com
- .bat
- .cmd

If your UPS does not have a low battery signal, you need to set the fields in the **UPS Characteristics** group box. The documentation from your UPS vendor should specify the battery recharge rate.

At startup, the UPS service assumes that the battery has no charge. If the battery recharge rate is 100 minutes per minute of run time, the computer has to run for 100 minutes before Windows NT expects the battery to have one minute of life.

Signals Used by the UPS Service

Windows NT uses a serial (COM) port to connect to the UPS. Table 5.2 shows the signals that the UPS service uses to communicate with the UPS, the associated pin numbers, and the function each signal provides to the UPS service. You need to know which of these signals your UPS supports so that you can configure the UPS service.

If you use a UPS service provided by your UPS vendor, the signals and their usage will probably be different than the signals that the Windows NT UPS service uses.

Table 5.2 UPS serial port connections

RS-232 Signal	9 Pin Connector Pin #	25 Pin Connector Pin #	UPS Usage
CTS	8	5	Power failure
DCD	1	8	Low battery
DTR	4	20	UPS shutdown

In the **UPS Configuration** group box, check the boxes that correspond to the signals that your UPS supports. For each of these signals, the **UPS Interface Voltages** (negative or positive) indicates the voltage present when that event occurs.

Low Battery Signal

When the UPS battery begins to run low, the UPS uses this signal to notify the UPS service.

When fully charged, UPSs typically deliver from five to 20 minutes of DC power after AC power has failed. If your UPS can detect when the battery is running down, you should check the low battery signal box in the **UPS Configuration** group box.

If you do not check the low battery signal box, the UPS service uses the information in the **UPS Characteristics** group box to estimate how much time remains on the battery after a power failure.

Remote UPS Shutdown

You want to conserve the life of the UPS battery for as long as possible. The Windows NT UPS service uses the UPS Shutdown signal to notify the UPS that Windows NT has completed its orderly shutdown and that the UPS can shut off power to the computer.

Power Failure Signal

The power failure signal is generated by the UPS to signal the Windows NT UPS service that a power failure has occurred. The Windows NT processing that occurs when it receives this signal depends upon whether the UPS can send a low battery signal.

If the UPS can send a low battery signal:

- The Windows NT UPS service waits the amount of time specified in the **Time between power failure and initial warning message** field of the **UPS Service** group box. It then sends a message to all users connected to the computer running Windows NT telling them that a power failure has occurred. No more users can connect to the computer after the message is sent.
- If a signal is received from the UPS indicating that power has been restored before the low battery signal occurs, normal operations are resumed. Users are again allowed to connect to the computer.
- When the UPS signals a low battery condition, the file specified in the **Execute Command File** group box is executed. After execution of this file, Windows NT initiates shutdown.
- Once the shutdown is complete, the UPS service signals the UPS that battery power can be turned off if you have checked **Remote UPS Shutdown**.

If the UPS cannot send a low battery signal:

- The UPS service computes the expected battery life left by dividing the **Battery recharge time per minute of run time** (in the **UPS Characteristics** group box) into the amount of time that Windows NT has been running. If the result is less than two minutes, then the file specified in the **Execute Command File** group box is executed and allowed 30 seconds to complete. Windows NT then initiates a shutdown.
- If the expected battery life is greater than two minutes, the UPS service waits the amount of time specified in the **Time between power failure and initial warning message** field of the **UPS Service** group box. It then sends a message to all users connected to the computer running Windows NT telling them that a power failure has occurred. No more users can connect to the computer after the message is sent.

- If a signal is received from the UPS indicating that power has been restored while the expected battery life is still greater than two minutes, normal operation resumes. Users are again allowed to connect to the computer.
- When the calculated battery life reaches two minutes, the file specified in the **Execute Command File** group box is executed. After execution of this file, Windows NT initiates shutdown. If the program takes longer than 30 seconds to complete, Windows NT terminates it, and final shutdown take places.
- Once the shutdown is complete, the UPS service signals the UPS that battery power can be turned off if you have checked **Remote UPS Shutdown.**

Using the UPS Service

You should use the following Windows NT services in combination with the Windows NT UPS service:

- Alerter
- Messenger
- EventLog

The Alerter service sends alerts to selected users. The Messenger service sends messages to your local computer and to other users on the network who are running Windows NT. All detected power fluctuations and power failures are recorded in the event log, along with UPS service start failures and shutdown initiations.

If you want the UPS service to start each time you start your computer, you need to modify the startup information.

▷ **To start the UPS service automatically**

1. On Control Panel, double-click **Services**.
2. On the **Services** dialog box, click **UPS**, then click **Startup**.
3. On the UPS **Service** dialog box, select **Automatic**, and then click **OK**.

You might need to use the preceding procedure to set the other services to start automatically.

Testing the UPS

When the UPS service is started, it verifies the settings in the UPS dialog box by ensuring that the signal polarity on the CTS and DCD pins is opposite to that specified as the failure condition in the **UPS** dialog box. For example, if the **UPS** dialog box specifies that the UPS device supports a Power Failure Signal (CTS pin) with a positive signal, the UPS service checks to make sure that this pin is not already asserted positive (which would not happen unless you had started the system during a power failure).

This has some important implications. With an online UPS, the UPS device might shut itself off immediately if the configuration is incorrect. With a standby UPS, an incorrect configuration typically shuts the UPS device off as soon as a power failure is detected, effectively circumventing the purpose of the UPS. This is why it is important to configure and test your UPS device to ensure that it operates correctly.

On x86-based computers, there is one other reason that it is important to test your UPS. During startup, Windows NT sends a detection signal to each port in order to recognize hardware attached to that port. Some UPS units using serial monitoring implementations respond to the detection signal by turning off. If this happens, use the **/NoSerialMice** switch in the Boot.ini file to prevent the system from sending this signal to the COM port to which your UPS unit is connected.

You can use multiple /NoSerialMice switches. The format is **/NoSerialMice=[COM*x* | COM*x*,COM*y*,COM*z*,...]**

- If **/NoSerialMice** is specified without parameters, serial mouse detection is disabled on all the serial ports.
- If **/NoSerialMice=COM*x*** is specified, serial mouse detection is only disabled on COM*x*.
- If **/NoSerialMice=COM*x*,COM*y*,COM*z* ...** is specified, serial mouse detection is disabled on each of the specified ports.

To ensure that the computer is protected from power failures, you can test it by simulating a power failure (that is, by disconnecting the main power supply to the UPS device). Your computer and peripherals connected to the UPS device should remain operational, and messages should be displayed and events logged. Wait until the UPS battery reaches a low level to verify that a graceful shutdown occurs. Restore the main power to the UPS device, and check the event log to ensure that all actions were logged and that there were no errors.

Note Before you begin any UPS tests, make sure that your UPS has sufficient charge built up to run all of the devices connected to it for at least the duration of the test.

Simulating UPS Operation

The next figure shows a simple simulator that you can build to emulate the power failure and low battery signals from a UPS.

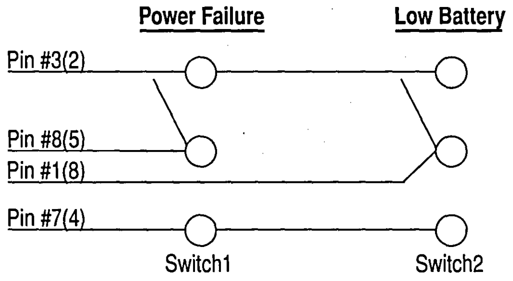


Figure 5.1. UPS simulator

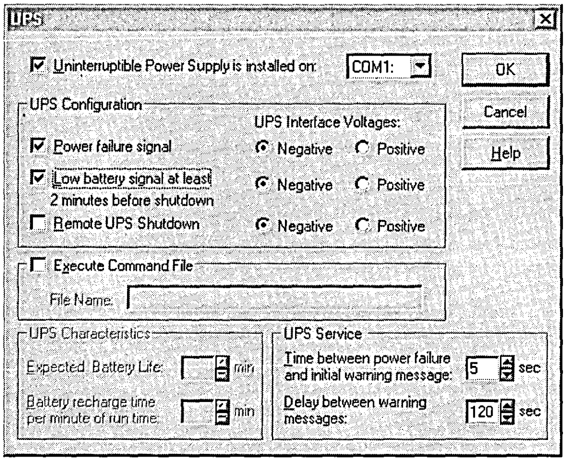
Note The pins numbers in the diagram are for a 9-pin serial port. For a 25-pin connector, use the pin numbers in parenthesis.

You can use two Single Pole Double Throw (SPDT) switches to construct the UPS simulator. To simulate normal operation, set both switches 1 and 2 to make contact with pin 7 (pin 4 on a 25-pin connector). Set both the **Power failure signal** and **Low battery signal at least 2 minutes before shutdown** to **Negative** in the **UPS** dialog box.

To simulate a power failure, move switch 1 to make contact with pin 3 (pin 2 on a 25-pin connector). Moving switch 2 to make contact with pin 3 simulates a low battery condition.

Simulating Power Failure and Low Battery Signals

To simulate the operation of a UPS with both a Power Failure and Low Battery signal, have both switches making contact with pin 7. Configure your UPS as shown in this dialog box:



UPS

☑ Uninterruptible Power Supply is installed on: COM1:

UPS Configuration

UPS Interface Voltages:

☑ Power failure signal ● Negative ○ Positive

☑ Low battery signal at least 2 minutes before shutdown ● Negative ○ Positive

☐ Remote UPS Shutdown ● Negative ○ Positive

☐ Execute Command File

File Name:

UPS Characteristics

Expected Battery Life: min

Battery recharge time per minute of run time: min

UPS Service

Time between power failure and initial warning message: 5 sec

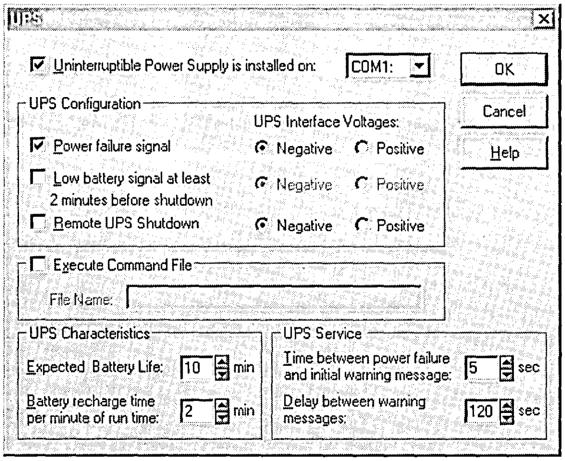
Delay between warning messages: 120 sec

Flip switch 1 to simulate a Power Failure signal. After five seconds, the initial warning message should be displayed. Every 120 seconds, the warning message should be displayed again.

When you know that the Power Failure signal processing is working correctly, flip switch 2. If you have checked the **Execute Command File** check box and specified a filename, the file should begin execution immediately. If the file does not finish executing within 30 seconds, Windows NT should terminate it. Windows NT should perform an orderly shutdown after the file terminates or immediately after the Power Failure signal if no file is specified.

Simulating UPS With Power Failure Signal Only

To simulate the operation of a UPS with only a Power Failure signal, have both switches making contact with pin 7. For purposes of this test, you can set the values in the **UPS Characteristics** group box to make it seem that your computer does not have to run very long before the battery is fully charged. When you set the values this way, you have to make sure that your battery does have enough charge to be able to complete the tests. For instance, if you set the values as shown in this dialog box, and let your computer run for 20 minutes, the UPS service assumes that the battery is fully charged.



☑ Uninterruptible Power Supply is installed on: COM1:

UPS Configuration

UPS Interface Voltages:

☑ Power failure signal ● Negative ○ Positive

□ Low battery signal at least 2 minutes before shutdown ● Negative ○ Positive

□ Remote UPS Shutdown ● Negative ○ Positive

□ Execute Command File

File Name:

UPS Characteristics

Expected Battery Life: 10 min

Battery recharge time per minute of run time: 2 min

UPS Service

Time between power failure and initial warning message: 5 sec

Delay between warning messages: 120 sec

You can test two scenarios with this configuration:

- If you connect switch 1 to pin 3 within four minutes of starting the UPS service, the expected battery life is less than two minutes, and Windows NT should begin shutdown immediately.
- If you allow the computer to run for 20 minutes before you flip switch 1, the battery is assumed to be fully charged. When you flip switch 1, the warning message should be displayed after five seconds. Every 120 seconds, another warning message should be displayed. Shutdown should not begin until the expected battery life decreases to two minutes.

CHAPTER 6

Troubleshooting Startup and Disk Problems

This chapter discusses what you can do to find the cause of problems when your computer fails to complete startup. Failure to complete startup means that the computer stops or displays an error message before you can log on to Windows NT. This chapter also discusses causes of and recovery from disk problems.

There are several utilities that you can use to troubleshoot these kinds of problems. The *Windows NT Server Resource Kit* CD contains a Windows NT-based utility, DiskProbe, that you can use to examine and change information on individual disk sectors. Chapter 7, "Disk, File System, and Backup Utilities," contains more information about DiskProbe and the MS-DOS-based utilities that you can use to read, write, and change information on disk sectors.

Other Sources of Information

There are several other sources of information for troubleshooting startup and disk problems:

- Chapter 8, "General Troubleshooting," contains details about Microsoft's sources for troubleshooting information and contains more details about troubleshooting hardware, software, and startup problems.
- The *Windows NT Workstation Resource Guide* contains several chapters that have troubleshooting information. These chapters are not included in the *Windows NT Server Resource Guide*:
    - Chapter 23, "Overview of the Windows NT Registry," describes how to use information in the Registry for troubleshooting and configuration maintenance.
    - Chapter 25, "Configuration Management and the Registry," provides problem solving techniques using the Registry.
    - Chapter 39, "Windows NT Debugger," describes the different kinds of Kernel STOP errors that you might see during startup. It also contains information about using the Windows NT Debugger.
- Appendix A, "Windows NT Setup Troubleshooting" in the *Windows NT Server Concepts and Planning* book, contains troubleshooting information, including answers to commonly asked questions.
- Windows NT online help contains a troubleshooting topic.
- The Microsoft Knowledge Base contains support information developed by Microsoft Product Support Specialists. The Windows NT Knowledge Base is included on the *Windows NT Server Resource Kit* CD. It is also included on the:
    - Microsoft Developer Network (MSDN) CD
    - TechNet CD
    - Microsoft Internet FTP host, ftp.microsoft.com
    - Technical Support and Services category on the Web page at http://www.microsoft.com/server

Troubleshooting Startup Problems on x86-based Computers

There are several things that can prevent a computer from successfully completing startup. The first step in figuring out what may be causing a problem is to determine whether the problem is occurring before the operating system takes control. If you do not see the boot loader screen on an x86-based computer, the problem may be due to a hardware failure, the Master Boot Record, Partition Tables, or Partition Boot Sector, which might be damaged.

There are several ways such damage can happen, including viruses. On x86-based computers, viruses use BIOS INT (interrupt) 13 calls to install themselves, so they are operating-system independent. Windows NT traps BIOS INT 13 calls while it is running, but cannot protect itself when the computer is started from an MS-DOS floppy disk or is dual-booted by using MS-DOS.

The Knowledge Base contains several articles about protecting your computer from viruses and recovering from problems with viruses. The sections "Protecting Against Viruses and Trojan Horses," in Chapter 4, and "Managing Shared Resources and Resource Security" in the *Windows NT Server Concepts and Planning* book contains information about reducing the chances that viruses will infect your computer.

If a problem occurs after selecting Windows NT from the boot loader screen, files that are needed by the operating system might be missing or corrupt.

If you have installed new hardware or new drivers, they could be causing the problem.

Problem Occurs Before the Boot Loader Starts

This section describes the problems that might occur between the time you turn your computer on until you see the boot loader screen.

Symptoms of problems in this group are:

- The computer hangs immediately after the Power On Self Test (POST).
- You do not see the boot loader screen.
- You receive messages such as:
    - Missing operating system.
    - A disk read error occurred.
    - Insert a system diskette and restart the system.
    - Invalid partition table.
    - Hard Disk Error.
    - Hard Disk Absent/Failed.

It is possible that you will not be able to start your computer to troubleshoot the problem. If all of your primary partitions are NTFS, using MS-DOS-based utilities will not do much good. If you have created a Windows NT startup floppy disk, as described in Chapter 5, "Preparing for and Performing Recovery," you can try using that disk. If you cannot start the computer by using your Windows NT startup floppy disk, and repairing your system by using the Emergency Repair Disk does not fix the problem, you can try removing it from the computer and installing it as a second disk on another Windows NT computer. You can then use Windows NT-based utilities for troubleshooting.

Caution Moving disks between computers is not supported because of problems that can arise when the disk controllers on the two systems are incompatible or are configured differently. However, if your two computers are configured the same, you might be able to identify and correct the problem.

The problem could be one of the following:

- There is no system partition on the first hard disk.
- The Master Boot Record is corrupt.
- The Partition Boot Sector is corrupt.
- The Boot.ini file is missing.
- The Windows NT boot loader, NTLDR, is missing or corrupt.
- The Complementary Metal Oxide Semiconductor (CMOS) is corrupt, or the CMOS battery is run down.
- Hardware has malfunctioned.

Troubleshooting System Partition Problems

When you startup from the hard disk on x86-based computers, the system BIOS code identifies the boot disk (usually disk 0), and reads the Master Boot Record. The code in the Master Boot Record searches for a system partition on the hard disk. If it cannot find such a system partition, or cannot start Windows NT from it, the startup process stops. If you get an error message such as "Error loading operating system," then the Master Boot Record code found a system partition, but could not start the operating system.

It is possible that there is no system partition on the hard disk from which you want to startup the computer. You might also have the wrong partition identified as the system partition. You can use the MS-DOS-based utility Fdisk to look at the partition information. For information about other utilities that have similar functionality, see Chapter 7, "Disk, File System, and Backup Utilities."

Note The system partition is the primary partition on the boot disk (usually disk 0) that has the Boot Indicator field set to 0x80. It contains the files that are needed to load Windows NT, such as Boot.ini and NTLDR.

Fdisk refers to the system partition as the active partition.

▷ Using Fdisk to check for and set the system partition

1. Start MS-DOS, or start from an MS-DOS startup floppy disk that contains the Fdisk utility and type **fdisk**.

   Fdisk displays the following message if there is no system partition on your first hard disk:

   ```
   WARNING!  No partitions are set active - disk 1 is not startable
   unless a partition is set active.
   ```

2. The FDISK Options screen has several choices. The cursor will be on Enter choice. Type 2 (**Set Active partition**). Fdisk displays information about the partitions on the hard disk. One partition should have an **A** in the **Status** column, which indicates it is the (active) system partition.
3. If there is no system partition, or the wrong partition is set as the system partition, enter the number of the partition that contains the files to use when loading the operating system. Fdisk displays the message Partition X made active, where X is the partition number you entered.
4. Press ESC to return to the **FDISK Options** menu, and press ESC again to exit Fdisk. You can now restart the computer. Be sure to remove the floppy disk.

Not all computers should be set to start from the first primary partition. For example, you can configure multiple partitions and install different operating systems on each partition. Some computers have EISA configuration partitions, and normally start from the second physical partition. For example, many Compaq computers are configured this way. However, the system partition should always be on your first physical hard disk.

Troubleshooting Master Boot Record Problems

This section describes troubleshooting Master Boot Record problems. The section titled "Master Boot Record," within Chapter 3, "Disk Management Basics," contains details about the Master Boot Record.

The functions of the Master Boot Record code are to:

- Read the Partition Table entries in the same sector.
- Determine the location of the Partition Boot Sector.
- Load and execute the code in the Partition Boot Sector.

If the executable code in the Master Boot Record does not do these functions, it displays one of these error messages:

- Missing operating system.
- Invalid partition table.

Note There is a Master Boot Record on each hard disk. However, only the Master Boot Record on the first hard disk is used in starting Windows NT.

Use DiskProbe or an MS-DOS-based utility to display the Master Boot Record. The example here shows what you should see at Cylinder 0, Side 0, Sector 1, which is the location for the Master Boot Record. This example shows the executable code in a Master Boot Record. This example might not match the code in the Master Boot Record on your computer, because some third-party boot and disk partitioning utilities modify the code in the Master Boot Record.

```
Physical Sector: Cyl 0, Side 0, Sector 1
00000000: 00 33 C0 8E D0 BC 00 7C - 8B F4 50 07 50 1F FB FC   .3.....|..P.P..
00000010: BF 00 06 B9 00 01 F2 A5 - EA 1D 06 00 00 BE BE 07   ................
00000020: B3 04 80 3C 80 74 0E 80 - 3C 00 75 1C 83 C6 10 FE   ...<.t..<.u.....
00000030: CB 75 EF CD 18 8B 14 8B - 4C 02 8B EE 83 C6 10 FE   .u......L.......
00000040: CB 74 1A 80 3C 00 74 F4 - BE 8B 06 AC 3C 00 74 0B   .t..<.t.....<.t.
00000050: 56 BB 07 00 B4 0E CD 10 - 5E EB F0 EB FE BF 05 00   V.......^.......
00000060: BB 00 7C B8 01 02 57 CD - 13 5F 73 0C 33 C0 CD 13   ..|...W.._s.3...
00000070: 4F 75 ED BE A3 06 EB D3 - BE C2 06 BF FE 7D 81 3D   Ou...........}.=
00000080: 55 AA 75 C7 8B F5 EA 00 - 7C 00 00 49 6E 76 61 6C   U.u.....|..Inval
00000090: 69 64 20 70 61 72 74 69 - 74 69 6F 6E 20 74 61 62   id partition tab
000000A0: 6C 65 00 45 72 72 6F 72 - 20 6C 6F 61 64 69 6E 67   le.Error loading
000000B0: 20 6F 70 65 72 61 74 69 - 6E 67 20 73 79 73 74 65    operating syste
000000C0: 6D 00 4D 69 73 73 69 6E - 67 20 6F 70 65 72 61 74   m.Missing operat
000000D0: 69 6E 67 20 73 79 73 74 - 65 6D 00 00 80 45 14 15   ing system...E..
000000E0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
000000F0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000100: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000110: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000120: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000130: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000140: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000150: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000160: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000170: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000180: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
00000190: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
000001A0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
000001B0: 00 00 00 00 00 00 00 00 - FD 4E F2 14 00 00 80 01   .........N......
000001C0: 01 00 06 0F 7F 96 3F 00 - 00 00 51 42 06 00 00 00   ....•.?...QB....
000001D0: 41 97 07 0F FF 2C 90 42 - 06 00 A0 3E 06 00 00 00   A....,.B...>....
000001E0: C1 2D 05 0F FF 92 30 81 - 0C 00 A0 91 01 00 00 00   .-....0.........
000001F0: C1 93 01 0F FF A6 D0 12 - 0E 00 C0 4E 00 00 55 AA   ...........N..U.
```

There are two things that you can check in the Master Boot Record:

- The error message text should be as shown in the example.
- There should be only zeroes between the text Missing operating system and location hex 1B8.

A disk signature might or might not be present starting at location 1B8. Having no disk signature does not necessarily indicate a problem.

▷ **Using DiskProbe to display the Master Boot Record**

1. Click the DiskProbe icon in the Resource Kit folder.
2. On the **Drives** menu, click **Physical Drive**. The **Available Physical Drives** are listed as PhysicalDrive*x*, where x=0 for the first hard disk. Double click the disk that contains the Master Boot Record used to start the computer. In the case of an x86-based computer, this disk is usually PhysicalDrive0.
3. In the **Handle 0** group box, click **Set Active**. Click **OK**.
4. On the **Sectors** menu, click **Read**. Set **Starting Sector** to 0 and **Number of Sectors** to 1. Click **Read**.

Troubleshooting Partition Boot Sector Problems

Several known viruses can cause problems with the Partition Boot Sector, even if the volume is formatted with the NTFS file system. Infection can occur by running an MS-DOS-based program from either a floppy disk or by starting up MS-DOS on a dual-boot computer. Most viruses use BIOS INT 13 calls to transfer themselves to an absolute sector on the disk. Windows NT cannot protect itself from this type of infection when it is not running.

In some cases, the damage to the Partition Boot Sector can cause the computer to stop after displaying a blue screen with the message STOP 0x0000007B INACCESSIBLE_BOOT_DEVICE. Another symptom of a Partition Boot Sector problem is that the computer stops before displaying any messages, and the screen remains black.

▷ **Using DiskProbe to display the Partition Boot Sector**

1. Use the DiskProbe procedure in "Troubleshooting Master Boot Record Problems," presented earlier in this chapter, to read the Master Boot Record, which contains the Partition Table.
2. On the **View** menu, click **Partition table**. In the **Partition table index** list box, double click the partition number whose Partition Boot Sector you want to read.
3. Click the **Go** button next to **Relative Sector**.
4. On the **View** menu, click **Bytes** to see a display of the Partition Boot Sector in hex. Otherwise, click **NTFS BootSector** or **FAT BootSector** to see a formatted display of the information.

Because the Partition Boot Sector contains several fields that are computer-specific, every byte in your Partition Boot Sector will not be identical to the information shown in the examples here. The following information should be the same:

- The first three bytes should be the jump instruction.
- The next 11 bytes should be the OEM header string.
- There should be error message text toward the end of the sector.

These examples show portions of the Partition Boot Sector that should be the same or contain similar text on all computers. The first example is for a FAT volume, the second one is for an NTFS volume formatted when running Windows NT 3.51, and the third one is for an NTFS volume formatted when running Windows NT 4.0.

```
00000000: EB 3C 90 4D 53 44 4F 53 - 35 2E 30 00 02 08 01 00   .<.MSDOS5.0.....
00000180: 90 A2 07 02 F8 CB 42 4F - 4F 54 3A 20 43 6F 75 6C   ......BOOT: Coul
00000190: 64 6E 27 74 20 66 69 6E - 64 20 4E 54 4C 44 52 0D   dn't find NTLDR.
000001A0: 0A 00 42 4F 4F 54 3A 20 - 49 2F 4F 20 65 72 72 6F   ..BOOT: I/O erro
000001B0: 72 20 72 65 61 64 69 6E - 67 20 64 69 73 6B 0D 0A   r reading disk..
000001C0: 00 50 6C 65 61 73 65 20 - 69 6E 73 65 72 74 20 61   .Please insert a
000001D0: 6E 6F 74 68 65 72 20 64 - 69 73 6B 00 4E 54 4C 44   nother disk.NTLD
000001E0: 52 20 20 20 20 20 20 00 - 00 00 00 00 00 00 00 00   R      .........
000001F0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 55 AA   ..............U.
```

```
00000000: EB 5B 00 4E 54 46 53 20 - 20 20 20 00 02 01 00 00   .[.NTFS    .....
00000130: F2 C3 1D 00 41 20 64 69 - 73 6B 20 72 65 61 64 20   ....A disk read 
00000140: 65 72 72 6F 72 20 6F 63 - 63 75 72 72 65 64 2E 0D   error occurred..
00000150: 0A 00 29 00 41 20 6B 65 - 72 6E 65 6C 20 66 69 6C   ..).A kernel fil
00000160: 65 20 69 73 20 6D 69 73 - 73 69 6E 67 20 66 72 6F   e is missing fro
00000170: 6D 20 74 68 65 20 64 69 - 73 6B 2E 0D 0A 00 25 00   m the disk....%.
00000180: 41 20 6B 65 72 6E 65 6C - 20 66 69 6C 65 20 69 73   A kernel file is
00000190: 20 74 6F 6F 20 64 69 73 - 63 6F 6E 74 69 67 75 6F    too discontiguo
000001A0: 75 73 2E 0D 0A 00 33 00 - 49 6E 73 65 72 74 20 61   us....3.Insert a
000001B0: 20 73 79 73 74 65 6D 20 - 64 69 73 6B 65 74 74 65    system diskette
000001C0: 20 61 6E 64 20 72 65 73 - 74 61 72 74 0D 0A 74 68    and restart..th
000001D0: 65 20 73 79 73 74 65 6D - 2E 0D 0A 00 00 00 00 00   e system........
000001E0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00   ................
000001F0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 55 AA   ..............U.
```

```
00000000: EB 5B 90 4E 54 46 53 20  - 20 20 20 00 02 01 00 00   .[.NTFS    .....
00000130: 07 00 CD 10 EB F2 C3 1D  - 00 41 20 64 69 73 6B 20   .........A disk
00000140: 72 65 61 64 20 65 72 72  - 6F 72 20 6F 63 63 75 72   read error occur
00000150: 72 65 64 2E 0D 0A 00 29  - 00 41 20 6B 65 72 6E 65   red....).A kerne
00000160: 6C 20 66 69 6C 65 20 69  - 73 20 6D 69 73 73 69 6E   l file is missin
00000170: 67 20 66 72 6F 6D 20 74  - 68 65 20 64 69 73 6B 2E   g from the disk.
00000180: 0D 0A 00 25 00 41 20 6B  - 65 72 6E 65 6C 20 66 69   ...%.A kernel fi
00000190: 6C 65 20 69 73 20 74 6F  - 6F 20 64 69 73 63 6F 6E   le is too discon
000001A0: 74 69 67 75 6F 75 73 2E  - 0D 0A 00 33 00 49 6E 73   tiguous....3.Ins
000001B0: 65 72 74 20 61 20 73 79  - 73 74 65 6D 20 64 69 73   ert a system dis
000001C0: 6B 65 74 74 65 20 61 6E  - 64 20 72 65 73 74 61 72   kette and restar
000001D0: 74 0D 0A 74 68 65 20 73  - 79 73 74 65 6D 2E 0D 0A   t..the system...
000001E0: 00 17 00 5C 4E 54 4C 44  - 52 20 69 73 20 63 6F 6D   ...\NTLDR is com
000001F0: 70 72 65 73 73 65 64 2E  - 0D 0A 00 00 00 00 55 AA   pressed.......U.
```

Although other corruption problems are possible, if any of the strings are incorrect or missing, you should assume the sector is corrupt.

Even if there is no obvious damage, the Partition Boot Sector might not be working correctly. One way to check for this problem is to change the name of NTLDR to anything else and restart Windows NT from the hard disk.

You should see this error message on a FAT primary partition:

```
Couldn't find NTLDR.
```

You should see this error message on an NTFS primary partition:

```
A kernel file is missing from the disk.
```

These errors indicate that the Partition Boot Sector is okay and that the problem is with a corrupt NTLDR file. In this case, see "Replacing Windows NT Files on the System Partition" within Chapter 5, "Preparing For and Performing Recovery," for information about replacing NTLDR.

If you don't get this error message, then the problem is probably a corrupt Partition Boot Sector. To replace the sector, see "Replacing the Partition Boot Sector," within Chapter 5, "Preparing for and Performing Recovery." Be sure to change the name back to NTLDR so that the Partition Boot Sector can find it again.

NTLDR usually has the Hidden, System, and Read Only attributes set. Because you cannot start Windows NT when the Partition Boot Sector on the boot partition is corrupt, you can start the MS-DOS operating system from the MS-DOS startup floppy disk to change the attributes. To change the attributes by using the MS-DOS attribute command, enter:

```
attrib -s -h -r ntldr
```

CMOS Problems

The CMOS typically stores information about the following:

- Time and date
- Floppy disks
- Video type
- Hard disks
- Memory installed

Each manufacturer and BIOS vendor can decide what a user should be able to configure, and what the standard configuration is. You can access the CMOS through either a utility or a keyboard sequence, depending on the manufacturer. You should write down or use a utility to print all of the CMOS information.

The computer uses the CMOS checksum to determine if any CMOS values have been changed other than by using the CMOS Setup program. If the checksum is not correct, the computer will not start.

Once the CMOS is correctly configured, CMOS problems are usually caused by one of the following:

- A weak battery, which can happen if the computer has not been powered on for a long time.
- The connection between the CMOS and the battery is loose or faulty.
- The CMOS has been damaged from static electric discharge.

All of these problems can result in information in the CMOS being set to zero or otherwise corrupt, this halting the startup.

Hardware Problems

If a device fails to initialize during POST, there can be a problem with accessing it. If you have changed or added a device since the last startup, the problem might be with the new configuration.

If you have changed your disk configuration, you should check that:

- SCSI devices are terminated properly.
- The BIOS is enabled on only the first SCSI controller (if at all). See "Understanding ARC Pathnames" within Chapter 5, "Preparing For and Performing Recovery," for information about whether the controller BIOS should be enabled.
- There are no IRQ conflicts.

If you have not made any changes, check that

- Controller cards are seated properly.
- Cables are properly connected.
- Disks are all powered up.

Chapter 8, "General Troubleshooting," discusses hardware problems. The Knowledge Base also contains information about troubleshooting these kinds of problems.

Problem Occurs After the Boot Loader Starts

This section describes troubleshooting problems that might occur from the time NTLDR starts executing through the time that you successfully log onto Windows NT. This phase of startup begins when you see the following message:

```
NTDETECT V1.0 Checking Hardware . . .
```

Using Checked Version of NTDETECT

On x86-based computers, NTDETECT detects installed hardware components.

There is a debug version of Ntdetect.com on the *Windows NT Server* product CD, called Ntdetect.chk. If Ntdetect.com fails to detect all of the hardware that you think it should find, you can use the debug version to help isolate the problem. A mouse or a disk controller are the components that typically causes problems.

To use the checked version:

- Rename Ntdetect.com to Ntdetect.bak in the root folder of your system partition.
- Copy Ntdetect.chk from Support\Debug\I386 to the root folder.
- Rename Ntdetect.chk to Ntdetect.com.

The utility Installed on the *Windows NT Server Resource Kit* CD performs the same functions.

Ntdetect.com has attributes of Hidden, System, and Read Only set when you install Windows NT. You need to clear these attributes to make the file visible. You can change the attributes by using **My Computer**, **Windows NT Explorer**, or the command prompt.

▷ **To change file attributes by using My Computer or Windows NT Explorer**

1. On the **View** menu, click **Options**.
2. On the **View** tab, select **Show all files**. Click **OK**.
3. Click the filename NTLDR at the root of the C drive.
4. On the **File** menu, click **Properties**.
5. In the **Attributes** box of the **General** tab, clear the **Read Only** and **Hidden** check boxes, and then click **OK**.

To change the attributes by using the command prompt, enter:

```
attrib -s -h -r ntldr
```

Shutdown Windows NT and restart the computer. When Ntdetect.chk executes, it displays information on the screen as it detects the hardware. This is a sample of the kind of information you might see:

```
Detecting System Component . . .
Reading BIOS date . . .
Done reading BIOS date (1/20/1994)
Detecting Bus/Adapter Component . . .
Collecting Disk Geometry . . .
Detecting Keyboard Component . . .
```

After it finishes displaying information about the components, press ENTER so Ntdetect.chk will continue. Ntdetect.chk next displays information about the current nodes for the controllers and peripherals. You need to press ENTER at the end of each screen of information.

When you have finished using Ntdetect.chk, you should rename Ntdetect.com to Ntdetect.chk and rename Ntdetect.bak to Ntdetect.com.

Using the /maxmem Switch

For x86-based computers, the Boot.ini file has a /maxmem switch that enables you to specify the maximum amount of RAM memory that Windows NT can use. You can use this switch to troubleshoot memory parity errors, mismatched SIMM speeds, and other memory-related problems. To use this switch, the memory must be contiguous. You should never specify a value less than 12 for Windows NT Server.

Note Windows NT Server can run with 12 MB RAM, although it will probably run quite slowly.

You include this switch at the end of the ARC path specified in the **[operating systems]** section of the Boot.ini file. This example restricts Windows NT Server to using only the first 12 MB RAM.

```
multi(0)disk(0)rdisk(0)partition(1)\winnt=Windows NT Server 4.0
/MAXMEM=12
```

Chapter 8, "General Troubleshooting," contains more information about troubleshooting memory problems.

Using the /sos Switch

You can add the /sos switch to the Boot.ini file to have NTLDR display the Kernel and driver names while they are being loaded. Use this switch if Windows NT does not startup and you think a driver is missing or corrupted. See the section "Boot.ini Switches" within Chapter 3, "Disk Management Basics," for information about changing Boot.ini switches.

These examples assume that you have installed Windows NT Server in the folder winnt40 on partition 2.

First, you should see this message:

```
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\ntoskrnl.exe
```

And then this one:

```
Press spacebar now to invoke Hardware Profile/Last Known Good Menu
```

Regardless of whether you press the spacebar, you should see these two messages:

```
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\hal.dll
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\config.sys
```

The **Hardware Profile/Last Known Good** menu displays at this time if you pressed the spacebar or you have more than one hardware profile. After NTLDR finishes processing the hardware profile information, it clears the screen and displays information such as:

```
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\c_1252.nls
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\c_437.nls
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\n_intl.nls
multi(0)disk(0)rdisk(0)partition(2)\winnt40\FONTS\vgaoem.fon
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\Drivers\Atdisk.sys
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\Disk.sys
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\CLASS2.SYS
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\Diskperf.sys
multi(0)disk(0)rdisk(0)partition(2)\winnt40\System32\Ntfs.sys
```

Fatal System Error: 0x0000006B

In some cases on ESDI disks with more than 1024 cylinders, Windows NT Server appears to have been successfully installed. However, the first time that you attempt to start from the hard disk, NTLDR loads various files and then produces a Fatal System Error: 0x0000006b with the message that Phase 1 Process Initialization failed. Following this message is a hexadecimal dump and system lockup. If you experience this problem, read the section "ESDI Hard Disk" within Chapter 3, "Disk Management Basics," for information about the ESDI controllers that Microsoft has tested.

Stop 0x0000007B — Inaccessible Boot Device

This STOP message means that Windows NT is unable to access the Partition Boot Sector or the required information is not found. A common cause of this error is a virus.

Another cause of the problem is incompatible Logical Block Addressing (LBA). The system BIOS is designed to allow access to fixed disks that use fewer than 1024 cylinders. Many modern disks, however, exceed 1024 cylinders. LBA is used to provide support for these disks. Such support is often built into the system BIOS. However, there are potential problems with LBA:

- If partitions were created and formatted with LBA disabled, and LBA is subsequently enabled, a STOP 0x0000007B can result. The partitions must be created and formatted while the LBA is enabled. In addition, changing LBA modes from one scheme to another can cause you to have to recreate the partitions and reformat the disk.
- Some LBA schemes are not compatible with Windows NT. When in doubt, it is best to check with your vendor.

A corrupt Partition Boot Sector can also cause this STOP message, depending upon what part is corrupt. This problem is similar to a virus, except that the corruption is caused by such things as a defective disk or controller, or a bug in a program that somehow has corrupted the Partition Boot Sector.

Troubleshooting Startup Problems on RISC-based Computers

RISC-based computers generally have fewer problems at startup than x86-based computers. This is because they usually do not have problems with viruses destroying the Master Boot Record or the Partition Boot Sector. They also do not have problems with the disk configuration information in the computer not matching what the controller is using.

The startup problems common to RISC-based computers are:

- Missing or incorrect file(s). This problem is usually the result of a user deleting, moving, or renaming a file that is needed for startup. It is also possible that a device driver is not the correct version for the device.
- Incorrect firmware. Upgrading to a new version of Windows NT typically requires upgrading the firmware. The firmware upgrades are available through your RISC-based computer vendor.

On RISC-based computers, you can use the /sos switch to display the Kernel and device driver names while they are being loaded. Include this switch on the OSLOADOPTIONS variable in the firmware. See the section "Using the /sos Switch," presented earlier in this chapter, for a description of the output you see when using this switch. There is information about editing the firmware variables in the section titled "Manage Boot Selection Menu," within Chapter 19, "What Happens When You Start Your Computer," in the *Windows NT Workstation Resource Guide*.

Troubleshooting problems with RISC-based computers can be more difficult than troubleshooting problems with x86-based computers because there are few disk or hardware troubleshooting tools available outside of Windows NT. If you can startup Windows NT, which you might be able to do by using your Windows NT startup floppy disk, try using these utilities:

- The Windows NT Diagnostics in the Administrative Tools program group, which enables you to look at hardware information.
- The DiskProbe utility on the *Windows NT Server Resource Kit* CD, which is a low-level disk editor that enables you to examine and change individual disk sectors.

On RISC-based computers, there are no recovery tools such as those available when you dual-boot MS-DOS on x86-based computers. If your startup problem is because of a problem with information on the hard disk, you might have to remove the disk from the RISC-based computer and install it as a second disk on an x86-based computer to troubleshoot the problem. When you use this approach, you can use a low-level disk editor to examine the information on the disk in the same way you would troubleshoot a problem with a disk on an x86-based computer. Although these kinds of problems do not occur very often, you might have to use this approach after a power failure or system crash.

Note On NTFS volumes, you need to use Windows NT-based utilities, such as DiskProbe, to examine information on the volume.

Troubleshooting Startup Problems Common to Both x86-based and RISC-based Computers

When starting your computer, Windows NT provides options for using alternative configurations. If you have changed hardware or device drivers since the last startup and Windows NT does not complete startup, try using one of the other options.

Troubleshooting by Using the Last Known Good Configuration

Windows NT provides two configurations in which you can start your computer:

- Default, the configuration that was saved when you shutdown the computer.
- Last Known Good, the configuration that was saved when you last logged on to your computer.

The configurations are stored as control sets in the Registry key HKEY_LOCAL_MACHINE\SYSTEM. If you made changes to your configuration when you were last logged on, such as adding drivers, changing services, or changing hardware, the two control sets will contain different information. As soon as you log on however, the information in these control sets will be the same. Therefore, if you are having problems with startup, and think the problems might be related to changes in your configuration, do not log on. Instead, shutdown the computer and restart it. Then, select **Last Known Good** from the **Hardware Profile/Last Known Good** menu to recover from the following types of problems:

- You install a new device driver, restart Windows NT, and the system stops responding. The Last Known Good control set enables you to startup because it does not contain any reference to the new, faulty driver.
- You install a new video driver and are able to start the system. However, you cannot see anything, because the new video resolution is incompatible with your video adapter. Do not try to log on. If you have the option to shutdown the computer without logging on, do so. If that option is not available, you need to restart your computer by turning it off or using the reset button. Wait for all disk activity to stop before you initiate the restart, especially if the computer has FAT volumes.
- You accidentally disable a critical device driver (such as the Scsiport driver). Windows NT is not be able to start and automatically reverts to the Last Known Good control set.

Using the Last Known Good control set does not help in the following situations:

- Any problem that is not related to changes in control set information, such as information like user profiles and file permissions.
- After you logged on after making changes. Here, the Last Known Good control set has already been updated to include the changes.
- Switching between different hardware profiles, such as docked and undocked laptops. The Last Known Good control set is only a method to switch between configuration information in the Registry. Use Hardware Profiles for this.
- Startup failures caused by hardware failures or corrupted files.
- Copying a new driver over the top of an old one, and the old one is already active.

Chapter 8, "General Troubleshooting," contains more details about troubleshooting by using the Registry key HKEY_LOCAL_MACHINE. For more information about hardware profiles and control sets, see Chapter 19, "What Happens When You Startup Your Computer," in the *Windows NT Workstation Resource Guide.*

Troubleshooting Video Problems

If your screen stays black or is skewed after a restart, either the video device is not resetting correctly during the restart or the video is sharing an IRQ.

Turn the power off and restart the computer. If the video works, you will probably need to turn the power off each time you restart Windows NT. This problem is video-and system-BIOS related.

If the video is still wrong after turning the power off and restarting, check for IRQ and memory conflicts with other cards on your system. If you are using a PCI-based computer, make certain that the video is not using IRQs 2, 9, or 12.

If you have installed a new video driver, or changed the display type by using the Display option in Control Panel, you might have created an incompatibility between the driver and the video device. One way that you can tell you have a problem is if you get a black screen instead of the logon message when you restart Windows NT.

You can turn off your computer or use the reset button to restart your computer. Then select the Last Known Good configuration. This recovery method is described in "Troubleshooting by Using the Last Known Good Configuration," presented earlier in this chapter.

On x86-based computers, you have another way to recover. When you install Windows NT, it creates two paths to the Windows NT folder, such as:

```
scsi(0)disk(0)rdisk(0)partition(1)\WINNT="Windows NT Server Version
4.00"
scsi(0)disk(0)rdisk(0)partition(1)\WINNT="Windows NT Server Version 4.00
[VGA mode]" /basevideo
```

If you select the path with the VGA mode option, Windows NT starts up by using the standard VGA driver. You can then use the Display option to reconfigure your video device.

Note Windows NT Server version 4.0 requires new video and printer drivers. Windows NT Server version 3.51 drivers for these devices will not work correctly when you are running Windows NT version 4.0.

Troubleshooting Disk Problems That Occur After Logon

This section discusses problems that occur after you have successfully logged on.

Volume is Displayed as Unknown

If you have created and formatted a volume with NTFS or FAT, but you can no longer access files on it, and Disk Administrator displays the volume as Unknown, the Partition Boot Sector for the volume might be bad.

For NTFS volumes, there are two other possible causes for this error:

- Permissions for the volume have been changed.
- The Master File Table (MFT) is corrupt.

The Partition Boot Sector can be corrupted by viruses. Corruption problems can also occur if you have a dual-boot configuration with Windows 95 and you use the Windows 95 version of Fdisk. To avoid problems with Fdisk, delete the Windows 95 version, make sure that you have the MS-DOS-based version, and run Fdisk only when you start MS-DOS.

To determine if a corrupt Partition Boot Sector is causing the problem, and for the description of the procedures to restore it, see “Replacing the Partition Boot Sector” within Chapter 5, “Preparing for and Performing Recovery.”

The permission problem occurs when you have done all the following:

- Created a second volume.
- Removed the group Everyone from the Access Control List (ACL).
- Granted access to a specific user.

The single user has normal access, but when other users log on, or if Windows NT is reinstalled, Disk Administrator sees the drive as unknown.

To correct this problem, log on as an administrator and take ownership of all folders, or add the group Everyone back with full control (default).

If there is corruption in the MFT file, there is no general solution. If you suspect this kind of problem, you should contact your technical support personnel for assistance.

Extended Partition Problems

If a Partition Table that defines a logical drive within an extended partition becomes corrupt, Windows NT can no longer access that volume, or the volumes that follow it on the disk. Once the pointer to the next volume has been lost, Windows NT can no longer find out where the volumes start.

When an extended partition becomes corrupt, it might be possible to rebuild it by using a sector editor or partition table editor.

For information about the organization of extended partitions, see "Logical Drives and Extended Partitions" within Chapter 3, "Disk Management Basics." For information about ways to attempt to repair the extended partition, see the description of the DiskProbe utility within Chapter 7, "Disk, File System, and Backup Utilities."

CHAPTER 7

Disk, File System, and Backup Utilities

This chapter explains various disk utilities, file system utilities, and backup utilities and how to use them. It shows where to obtain the utilities and other sources of related information.

Other chapters in this part of the *Windows NT Server Resource Guide* also identify situations in which you should use the utilities (described in this chapter) and contain more information about them:

- Chapter 3, "Disk Management Basics."
- Chapter 5, "Preparing for and Performing Recovery."
- Chapter 6, "Troubleshooting Startup and Disk Problems."

The utilities described in this chapter are available from different sources.

The following utilities are on the *Windows NT Server* product CD. When you install Windows NT, they are installed in *%systemroot%*\System32. You must run all of these utilities from the command prompt. The Section column in the next table identifies the section in this chapter in which the utility is described.

Utility	Purpose	Section
AT	Run Windows NT backup at scheduled time.	Backup
Cacls	Display and modify Access Control Lists (ACLs) for NTFS files and folders.	File System
Compact	Display and change compression of NTFS files and folders.	File System
Convert	Convert volume from FAT to NTFS.	File System
Expand	Expand files that are compressed on the Windows NT Server product CD, the *Windows NT Server Resource Kit* CD, or that you compressed by using the Compress utility.	File System

The rest of the utilities described in this chapter are on the *Windows NT Server Resource Kit* CD. You can install the Resource Kit by double-clicking Setup.exe when you load the Resource Kit CD.

The Windows NT Resource Kit utilities are organized into subgroups, by functional area. Rktools.hlp contains information about each of the utilities on the resource kit. To find out which functional area contains which utilities, double-click Rktools.hlp and double-click each Tools group on the Contents screen.

When you install the Windows NT Resource Kit, Setup creates a Resource Kit program group that has Rktools.hlp and Readme.txt in it. It also creates a program group within the Resource Kit program group for each subgroup that you install. You can install only those subgroups that you want by doing a custom install.

▷ To do a custom installation of the Resource Kit

1. Double-click Setup.exe at the root of the *Windows NT Server Resource Kit* CD.
2. Click the **Custom** button.
3. Check the group(s) that contains the utilities you want to install.

The following table contains information about each of the Resource Kit utilities described in this chapter. As in the previous table, the Section column identifies the section in this chapter in which the tool is described.

Utility	Type	Purpose	Subgroup	Section
Compress	Command prompt	Compress files and folders.	File	File System
DirUse	Command prompt	Display folder size information, including compression information for NTFS volumes.	File	File System
DiskMap	Command prompt	Display information about the disk and the contents of the Partition Table.	Disk/Fault Tolerance	Disk
DiskProbe	Windows NT-based	Low-level disk editor.	Disk/Fault Tolerance	Disk
DiskSave	MS-DOS-based	Save and restore Master Boot Record and Partition Boot Sector.	Disk/Fault Tolerance	Disk
Expndw32	Windows NT-based	Expand compressed files.	File	File System
FTEdit	Windows NT-based	Recover stripe sets, stripe sets with parity, and volume sets.	Disk/Fault Tolerance	Disk

Disk Utilities

DiskProbe, DiskSave, and FTEdit are low-level editors. When you use them, you are changing information on the disk without any checks as to whether the changes make sense. Each utility does display messages asking you to verify that you really want to change the information. However, you can easily make changes that have serious consequences, such as:

- You cannot start any operating system.
- A volume is no longer accessible.
- You have to recreate and reformat all of the partitions and logical drives.

Used carefully however, you can solve these same problems when they occur through human error, hardware problems, power outages, and similar events. It is a good idea to get familiar with using these utilities in a test situation before you need to use them for real. Testing is especially important if your configuration has volume sets, stripe sets, extended partitions, or extended volume sets.

DiskMap displays information layout of partitions and logical drives on your disk, and you cannot change information on the disk when using this utility.

Note You can save and restore Master Boot Records and Partition Boot Sectors by using both DiskProbe and DiskSave. Both utilities save these files in the same format, so they can be saved by using one utility and restored by using the other.

DiskMap — Display a Map of the Disk

This utility produces a report on the configuration of the disk that you specify. It provides information about the disk characteristics and a description for each partition and logical drive on the disk.

DiskMap can be run from the command prompt. The syntax is:

```
diskmap /d<drive#> [/h]
```

These are the options for DiskMap

Option	Description
/d<drive#>	Drive# is the physical disk for which you want a map.
/h	Indicates hexadecimal output. The default is decimal output.

Some fields in the DiskMap report are always hex or always decimal, regardless of whether you specify the **/h** option when you run the utility. These fields are shown in the next table. You can see the fields in the DiskMap sample report, shown later in this section.

Always hex	Always decimal
Signature	The MB portion of the DiskSize field
System ID	Partition number

As an example, Figure 7.1 is a Disk Administrator screen shot of a computer with one hard disk.

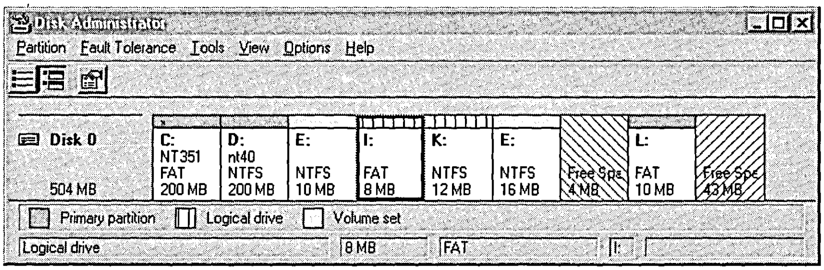


Figure 7.1 Disk Administrator screen shot

You get the following report for the disk configuration shown in Figure 7.1 when you enter **diskmap /d0** at the command prompt:

```
Cylinders  HeadsPerCylinder SectorsPerHead BytesPerSector MediaType
     1023                16             63            512        12
TrackSize = 32256, CylinderSize = 516096, DiskSize = 527966208 (503MB)

Signature = 0x14f24efd
     StartingOffset    PartitionLength StartingSector PartitionNumber
*             32256          210018816             63               1
          210051072          209534976         410256               2
          472227840           10321920         922320               3
          419618304           10289664             63               4
          429940224            8225280             63               5
          438197760           12354048             63               6
          450584064           16998912             63               7
```

```
MBR:
           Starting                    Ending         System    Relative   Total
  Cylinder Head Sector  Cylinder Head Sector    ID      Sector     Sectors
*        0    1      1       406   15     63   0x06          63     410193
       407    0      1       812   15     63   0x07      410256     409248
       813    0      1       914   15     63   0x05      819504     102816
       915    0      1       934   15     63   0x01      922320      20160

EBR: (sector 819504)
           Starting                    Ending         System    Relative   Total
  Cylinder Head Sector  Cylinder Head Sector    ID      Sector     Sectors
       813    1      1       832   15     63   0x87          63      20097
       833    0      1       848   15     63   0x05       20160      16128
         0    0      0         0    0      0   0x00           0          0
         0    0      0         0    0      0   0x00           0          0

EBR: (sector 839664)
           Starting                    Ending         System    Relative   Total
  Cylinder Head Sector  Cylinder Head Sector    ID      Sector     Sectors
       833    1      1       848   15     63   0x01          63      16065
       849    0      1       872   15     63   0x05       36288      24192
         0    0      0         0    0      0   0x00           0          0
         0    0      0         0    0      0   0x00           0          0

EBR: (sector 855792)
           Starting                    Ending         System    Relative   Total
  Cylinder Head Sector  Cylinder Head Sector    ID      Sector     Sectors
       849    1      1       872   15     63   0x07          63      24129
       873    0      1       905   15     63   0x05       60480      33264
         0    0      0         0    0      0   0x00           0          0
         0    0      0         0    0      0   0x00           0          0

EBR: (sector 879984)
           Starting                    Ending         System    Relative   Total
  Cylinder Head Sector  Cylinder Head Sector    ID      Sector     Sectors
       873    1      1       905   15     63   0x87          63      33201
         0    0      0         0    0      0   0x00           0          0
         0    0      0         0    0      0   0x00           0          0
         0    0      0         0    0      0   0x00           0          0
```

This utility is very useful for producing hardcopy reports for each disk, which should be kept with the other configuration information that you maintain for your computer. You can create a hardcopy by redirecting the output to a printer or to a file, which you can then print. This example creates a file.

```
diskmap /d0 >disk0map.txt
```

The first three lines of the output,

```
Cylinders  HeadsPerCylinder SectorsPerHead BytesPerSector MediaType
     1023                16              63             512      12
TrackSize = 32256, CylinderSize = 516096, DiskSize = 527966208 (503MB)
```

are information that DiskMap obtained by using IOCTL_DISK_GET_DRIVE_GEOMETRY. The information is the geometry of the disk as seen by Windows NT. The hardware might have different physical geometry, because translation can be performed either at the hardware or device driver layer. It is important to know what geometry Windows NT is using. You can use this information if you need to repair the Partition Table, which is described in the section entitled "Editing and Repairing the Partition Table," later in this chapter. The TrackSize, CylinderSize, and DiskSize values are bytes.

The second section of the output is information about the disk returned from IOCTL_DISK_GET_DRIVE_LAYOUT. Windows NT uses the signature to correlate information in the HKEY_LOCAL_MACHINE\SYSTEM\DISK Registry key with the appropriate physical disk when the disk contains volume sets or stripe sets. The StartingOffset and PartitionLength fields are the values for each of these data in bytes.

The MBR and EBR sections on the printout describe the contents of the Partition Table(s). The MBR section is for the Partition Table contained on the first sector of the disk. If there is no extended partition on the disk, there are no EBR sections. Otherwise, there is one EBR section for each logical drive in the extended partition. The fields in each of these sections are described in the "Partition Table" section within Chapter 3, "Disk Management Basics."

Each partition that has an asterisk (*) at the left side is a system partition, which contains the files used to load an operating system such as Windows NT. On an x86-based computer, each disk can have a system partition, but there should only be one partition with the * on each disk. RISC-based computers can have more than one system partition on a disk.

The DiskMap utility has several possible error messages. Some messages are not serious, and you can easily correct them. Other messages usually are caused by some type of hardware problem or corruption of the data on the disk. This table describes the error messages that you might see. The *N* and *E* in the message indicate numbers.

Message	Meaning
BLOCKED: CreateFile() Failed /d*N* [Error *E*] Ensure that you have selected a valid drive number and that the selected drive is not locked by another process.	The attempt to open physical disk *N* failed. Either some other program, such as Disk Administrator or DiskProbe, has the drive open, or the value following /d is invalid (out of range or not a number).
Unable to get drive layout using IOCTL_DISK_GET_DRIVE_GEOMETRY. Error: [Error *E*]. This is a fatal error.	The call to DeviceIoControl() with the parameter IOCTL_DISK_GET_DRIVE_GEOMETRY failed. You would get this error if you removed the disk in a removable media device. Otherwise, it could indicate a hardware failure where a device that was available at startup is no longer available.
Unable to allocate memory	DiskMap tried to allocate memory for a temporary buffer and no memory was available. Try closing other programs.
Unable to get drive layout using IOCTL_DISK_GET_DRIVE_LAYOUT. Error: [Error *E*].	The call to DeviceIoControl() with the parameter IOCTL_DISK_GET_DRIVE_LAYOUT failed. In attempting to walk the Partition Table chain, some kind of error was encountered. This error indicates Partition Table corruption, hardware RAID array corruption, hardware failure, or a change in apparent disk geometry that makes portions of the volume inaccessible. More information about why the call failed might be available in other parts of the utility where it walks the chain manually.

Message	Meaning
Unable to read next EBR. Error: [Error *E*].	A call to ReadFile() failed. This is usually caused by an attempt to read past the end of the disk, as in the case where the pointer to the next EBR is not valid. It can also indicate a bad sector at the EBR location.
Next EBR failed to pass sanity check.	This message means that the Ending Head value for each of the four Partition Table entries is not 0 or not equal to TracksPerCylinder-1, or that the last two bytes in the sector are not 0x55 0xAA. The problem is either an error in the chain of EBRs or this sector has been corrupted.
Detected partition corruption.	This message means that the value in the Relative Sector field for a Partition Table entry is not equal to the SectorsPerHead value.

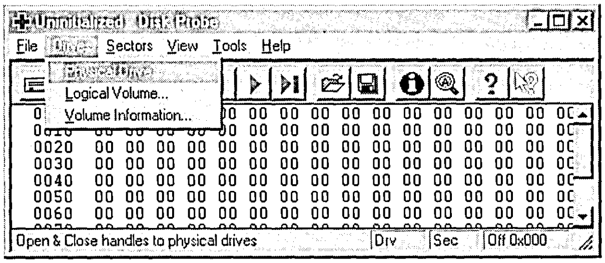
In general, if the partition size and layout displayed by DiskMap looks about right and the utility does not display any error messages, then the Partition Tables are probably correct.

An example of where DiskMap can be very useful is when a Partition Table was written incorrectly and the last EBR referenced is past the end of the disk. Disk Administrator will not initialize in this situation, because it depends on the information from IOCTL_DISK_GET_DRIVE_LAYOUT, which would fail. DiskMap would also display an error about IOCTL_DISK_GET_DRIVE_LAYOUT, but would display the EBRs that it can read up to the point of failure. If you have a hardcopy of a map of the disk that had no errors, you can see where the two chains diverged. You can restore the original partition information by correcting the first EBR that has bad values by using a low-level disk editor such as DiskProbe. Typically, the rest of the original EBRs would not have been damaged by the bad EBR chain. Sometimes, though, it might be necessary to repair more than one EBR.

DiskProbe — a Low-level Disk Editor

This Windows NT-based utility enables you to save, restore, find, examine, and change data on the disk. DiskProbe gives an administrator access to every byte on the physical disk without regard to access privilege, and with it you can change any part of the disk. It is your responsibility to make sure that you have backups of the disk.

After starting DiskProbe, as shown in Figure 7.2, you use the **Drives** menu to open a physical disk or a logical drive.



```
0                       0 00 00 00 00 00 00 00 00 00 00 00 0C
0010                    0 00 00 00 00 00 00 00 00 00 00 00 0C
0020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0C
0030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0C
0040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0C
0050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0C
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0C
```

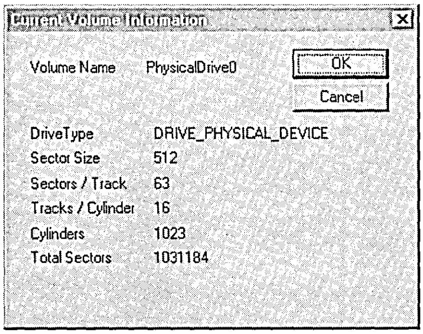
Open & Close handles to physical drives | Drv | Sec | Off 0x000

Figure 7.2 Opening a disk by using DiskProbe

Clicking **Physical Drive** on the **Drives** menu opens the entire physical disk. You can access any sector on the disk. When you read or write sectors, the sector number is the absolute sector from the beginning of the disk.

Clicking **Logical Volume** on the **Drives** menu opens the logical drive associated with a drive letter. Sector numbers are relative to the start of the logical drive, and you can only access sectors within the range of the logical drive. You cannot use DiskProbe to access a network drive.

When you have opened either a disk by clicking either **Physical Drive** or **Logical Volume**, you can get information about the characteristics of the disk by clicking **Volume Information** on the **Drives** menu. You get the same information regardless of how you opened the disk. This information is the same as what you see if you run the DiskMap utility for this disk. You can use this information if you need to repair the Partition Table. Here is an example:



Volume Name PhysicalDrive0

DriveType	DRIVE_PHYSICAL_DEVICE
Sector Size	512
Sectors / Track	63
Tracks / Cylinder	16
Cylinders	1023
Total Sectors	1031184

Whenever you use DiskProbe to read a sector, it displays the sector number of the first sector it read in the title bar. It also displays the sector number of the sector you are currently viewing in the status bar. This information is useful if you have read more than one sector.

When you are first using DiskProbe, or are looking at information in an extended partition, it is recommended that you write down the sector number for the start of each partition and logical drive. You should also write down the sector number of the Partition Boot Sector for each partition or logical drive. You can produce a hardcopy of the disk configuration by using the DiskMap utility, which is described earlier in this chapter.

If your disk has volume sets or an extended partition, remembering which primary partition or logical drive you are looking at can become difficult. You can always start over by reading sector 0, which reads the Partition Table for the disk.

Note In the DiskProbe utility, sectors are numbered starting from zero.

The location of some system information on NTFS volumes is different, depending on which version of Windows NT, was used to format the volume. For example, Windows NT 4.0 writes a backup of the Partition Boot Sector for the volume at the end of the volume. Earlier versions of Windows NT put the backup at the center of the volume.

For more information about DiskProbe, see Dskprobe.hlp and Rktools.hlp on the *Windows NT Server Resource Kit* CD.

Comparing DiskProbe to Disk Administrator

Although DiskProbe and Disk Administrator both provide you with information about the organization of your hard disks, each utility has additional functionality. Disk Administrator enables you to configure and format your disks. DiskProbe enables you to save and change the data on the disk.

Disk Administrator shows you, for each disk, the size and organization of:

- Primary partitions
- Logical drives (single contiguous areas in an extended partition)
- Stripe sets
- Volume sets
- Mirror sets
- Stripe sets with parity

When displaying information about stripe sets, volume sets, mirror sets, and stripe sets with parity, both DiskProbe and Disk Administrator read the Registry key HKEY_LOCAL_MACHINE\SYSTEM\DISK to determine which disks contain the members of the volumes.

When you view the Partition Table, DiskProbe enables you to determine the starting and ending location of primary partitions, logical drives, and volume sets.

DiskProbe simply displays the information that it finds in each Partition Table, it is up to you to understand how the individual entries in the Partition Table relate to what you have configured on your disks. If your configuration contains extended partitions, you might find it easier to understand the DiskProbe information if you print a screen shot of the Disk Administrator display (by pressing ALT+PRINT SCREEN, pasting the screen shot into Paint or a document, and then printing the picture or document). The section "Walking an Extended Partition," presented later in this chapter, contains an example of the Disk Administrator and DiskProbe information for an extended partition.

Note Because both DiskProbe and Disk Administrator can modify the Partition Table, and DiskProbe enables you to directly modify any sector on a disk, you should never have both of these utilities open at the same time. When you are using DiskProbe, you should also be very cautious about running any other programs. DiskProbe writes directly to the disk without using the file cache, the NTFS log file, or any file system device drivers.

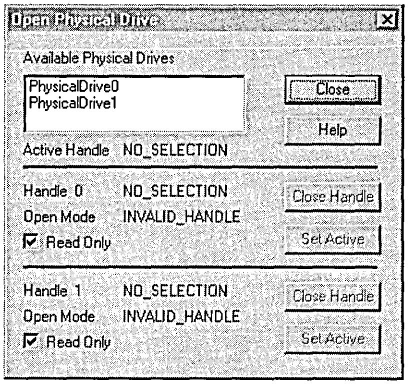
Backing Up and Restoring the Master Boot Record

The Master Boot Record on the hard disk that you use to start your computer is the most important sector on the disk. The Master Boot Record on other disks is not as critical. However, if the partition information in the other Master Boot Records is not correct, you might not be able to access volumes on those disks.

For an example of a Master Boot Record, see Chapter 3, "Disk Management Basics."

▷ To back up the Master Boot Record

1. In the Resource Kit folder, double-click the DiskProbe icon.
2. On the **Drives** menu, click **Physical Drive.** You will see the **Open Physical Drive** dialog box, which is shown in the following figure.



Open Physical Drive

Available Physical Drives
PhysicalDrive0
PhysicalDrive1

Active Handle NO_SELECTION

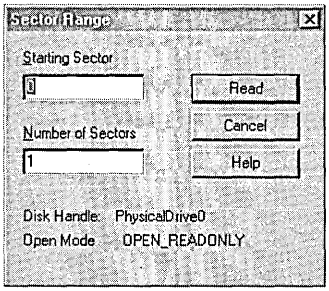
Handle 0 NO_SELECTION
Open Mode INVALID_HANDLE
☑ Read Only

Handle 1 NO_SELECTION
Open Mode INVALID_HANDLE
☑ Read Only

The **Available Physical Drives** are listed as PhysicalDrive*n*, where n=0 for the first hard disk. Double-click the disk that contains the Master Boot Record you want to save. You should always save the Master Boot Record on the startup disk. In the case of an x86-based computer, the startup disk is usually PhysicalDrive0.

When you have more than one disk, you should back up the Master Boot Record on each disk, since it contains the Partition Table for that disk.

3. In the **Handle 0** group box, click **Set Active.** Click **Close.**
4. On the **Sectors** menu, click **Read** to open the **Sector Range** dialog box, which is shown in the following figure. Set **Starting Sector** to 0 and **Number of Sectors** to 1. Click **Read.**



Sector Range

Starting Sector
0

Number of Sectors
1

Disk Handle: PhysicalDrive0
Open Mode OPEN_READONLY

Make sure that the sector you read looks like a Master Boot Record, as described within Chapter 3, "Disk Management Basics."

5. On the **File** menu, click **Save As**. Enter the filename. Consider saving the file to the Emergency Repair Disk, Windows NT startup floppy disk, or the MS-DOS bootable floppy disk. You can save it on more than one disk. If you have more than one disk whose master Boot Record you are backing up, using a filename such as Mbrdiskn.dsk can help you remember which file corresponds to which disk.

▷ **To restore the Master Boot Record**

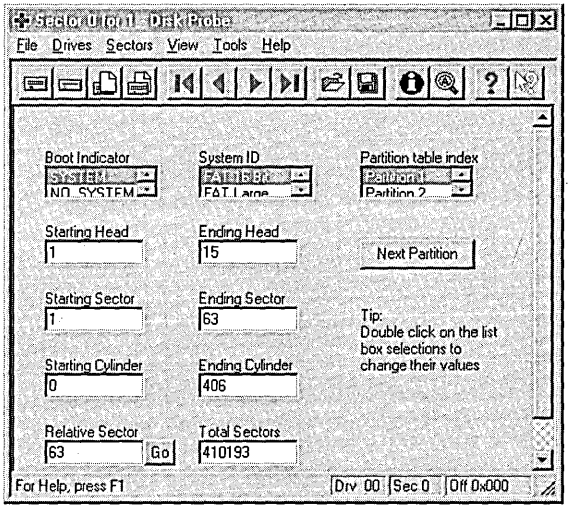
1. In the Resource Kit folder, double-click the DiskProbe icon.
2. On the **File** menu, click **Open**. Enter the filename and open the file. Verify that the file looks like a Master Boot Record and that it is 512 bytes. DiskProbe displays a message if the size of the file you open is not a multiple of 512.
3. On the **Drives** menu, click **Physical Drive**. The **Available Physical Drives** are listed as PhysicalDrive*n*, where n=0 for the first hard disk. Double-click the disk that contains the Master Boot Record you want to restore.
4. In the **Handle 0** group box, clear the **Read Only** check box, and click **Set Active**. Click **Close**.
5. On the **Sectors** menu, click **Write** to open the **Write Sector** dialog box. Set **Starting sector to write data** to 0. Click **Write it**.
6. DiskProbe displays a message box so you can verify the information.

Displaying and Using the Partition Table

The Master Boot Record is the first sector on each hard disk. The first step in troubleshooting Partition Table problems should be to read the Master Boot Record and look at the Partition Table. The description in this section applies to viewing the Partition Table in the Master Boot Record. See "Walking an Extended Partition," presented later in this chapter, for information about finding the Partition Table for an extended partition.

▷ To display the Partition Table

1. Read the Master Boot Record by using steps 1 through 4 in the procedure for backing up the Master Boot Record, given earlier in this chapter.
2. On the View menu, click Partition Table to see the information displayed in a user-friendly manner. You will see a screen similar to the following:



Boot Indicator: SYSTEM / NO_SYSTEM

System ID: FAT16Bit / FAT Large

Partition table index: Partition 1 / Partition 2

Starting Head: 1

Ending Head: 15

Starting Sector: 1

Ending Sector: 63

Starting Cylinder: 0

Ending Cylinder: 406

Relative Sector: 63

Total Sectors: 410193

Tip:
Double click on the list box selections to change their values

Drv 00 Sec 0 Off 0x000

Figure 7.3 DiskProbe Partition Table view

DiskProbe makes it easy for you to look at the information for each volume. In the **Partition table index** list box, double-click the partition number for which you want to see the Partition Table entry.

Click the **Next Partition** button to see the information in the first sector of the next partition. For a primary partition, clicking the **Next Partition** button reads the Partition Boot Sector of the next partition. When the next partition is an extended partition, clicking **Next Partition** reads the Partition Table sector for the first logical drive in the extended partition.

For a primary partition, clicking the **Go** button next to the **Relative Sector** field reads the Partition Boot Sector for the current partition. When the **System ID** field is EXTENDED, clicking the **Go** button reads the Partition Table for the next logical drive in the extended partition.

If you make changes to the Partition Table, you must explicitly write the sector to disk by clicking **Write** on the **Sectors** menu.

When you are viewing the information for an extended partition in the Partition Table area of the Master Boot Record, the **Total Sectors** field is the total size of the extended partition, which is usually larger than the size of the first logical drive in the partition.

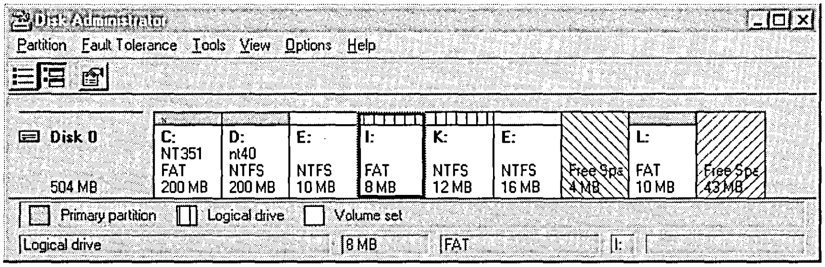
Walking an Extended Partition

If the disk has an extended partition, finding the information for each logical drive is more complicated than just looking at the Partition Table in the Master Boot Record, but DiskProbe does the work for you. "Logical Drives and Extended Partitions," within Chapter 3, describes the organization of an extended partition.

Information for the logical drives in an extended partition is basically chained from one logical drive to the next. The term *walking an extended partition* means that utilities like DiskProbe and DiskMap use the information defined for one logical drive in the chain to find the next logical drive in the chain. Knowing how to walk an extended partition is necessary if one of the links in the chain becomes corrupted, and you want to repair it.

The description in this section applies when you are viewing the Partition Table information in the first sector of each logical drive.

To help make the descriptions easier to understand, following is a screen shot from Disk Administrator, and the information from DiskProbe for the extended partition shown in the screen shot. The extended partition includes volume set E, logical drives I and K, and 4 MB of unused space between E and L. This example is the same disk configuration used in the section "DiskMap — Display a Map of the Disk," presented earlier in this chapter.



Disk 0 504 MB	C: NT351 FAT 200 MB	D: nt40 NTFS 200 MB	E: NTFS 10 MB	I: FAT 8 MB	K: NTFS 12 MB	E: NTFS 16 MB	Free Spa 4 MB	L: FAT 10 MB	Free Spa 43 MB

Logical drive | 8 MB | FAT | I:

Note This example is for illustration purposes, and is not a good organization for the disk, because the free space at the end of the primary partition is unusable. This is because you can have only four Partition Table entries in the Master Boot Record. In this example, primary partitions C, D, and L each use one entry. The extended partition, which includes volume set E, logical drives I and K, and the free space between E and K, uses the other entry. Disk Administrator displays an error message if you try to use any of the area after drive L to create a partition or extend an NTFS logical drive.

The distinction between the two areas of free space is that the one on the left is free space within an extended partition, while the free space on the right, which has a different background pattern, is not a part of any partition.

To walk the extended partition in this example, start with the information for Partition 3 in the Partition Table on relative sector 0. Partition 3 points you to the first logical drive in the extended partition. By following the links in the Partition Table entry for each logical drive, you can find the information for all of the logical drives.

The next table identifies which sectors contain information about the extended partition and the logical drives. The table also describes which buttons to click to walk the extended partition.

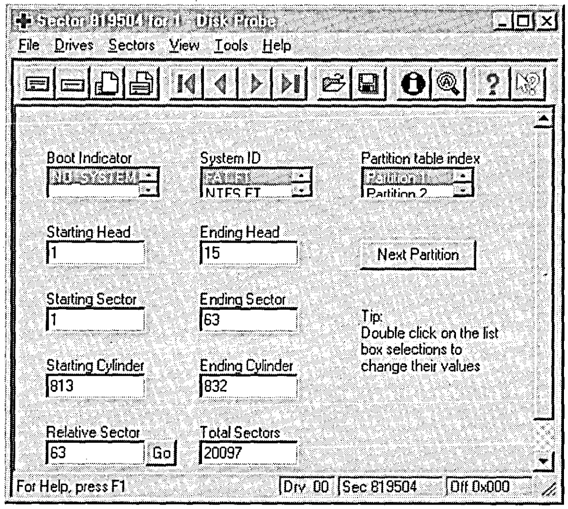
Relative sector	Contents
0	Partition 3 of the Master Boot Record contains an EXTENDED entry, which is the information about the first part of volume set E. Click **Go** when viewing Partition 3 to read the Partition Table for the first part of volume set E.
819504	Partition 1 describes the first part of volume set E. Partition 2 is an EXTENDED entry with the information for logical drive I. Click **Next Partition** when viewing Partition 1 to read the Partition Table for logical drive I.
839664	Partition 1 describes logical drive I, Partition 2 is an EXTENDED entry with the information for logical drive K. Click **Next Partition** when viewing Partition 1 to read the Partition Table for logical drive K.
855792	Partition 1 describes logical drive K. Partition 2 is an EXTENDED entry with the information for the second part of volume set E. Click **Next Partition** when viewing Partition 1 to read the Partition Table for the second part of volume set E.
879984	Partition 1 describes the second part of volume set E. Partition 2 is all zeroes (end of the extended partition).

The next table shows the information in Partition 3 of the Master Boot Record and each of the Partition 1 entries for the extended partition.

Field Name	**Master Boot Record entry**	**First part of E (10 MB)**	**I (8 MB)**	**K (12MB)**	**Second part of E (16 MB)**
System ID	EXTENDED	NTFS FT	FAT	NTFS	NTFS FT
Boot Indicator	NO_ SYSTEM	NO_ SYSTEM	NO_ SYSTEM	NO_ SYSTEM	NO_ SYSTEM
Start Head	1	1	1	1	1
End Head	15	15	15	15	15
Start Sector	1	1	1	1	1
End Sector	63	63	63	63	63
Start Cylinder	813	813	833	849	873
End Cylinder	914	832	848	872	905
Relative Sector	819504	63	63	63	63
Total Sectors	102816	20097	16065	24129	33201

When the **System ID** field for a Partition 2 in an extended partition contains the value EXTENDED, double-click Partition 1 in the **Partition table index** list box, and then click **Next Partition** to read the first sector of the next logical drive. The only information in the first sector of a logical drive is a Partition Table, which has two entries. The Partition 1 entry contains information about the current logical drive, and the Partition 2 entry is the information for the next logical drive in the extended partition. The **System ID** for each Partition 2 entry should be EXTENDED unless you are at the end of the extended partition.

The next screen shot shows the Partition Table entry for the first part of volume set E; the Partition Table is located at relative sector 819504.


Sector 819504 for 1 - DiskProbe
File Drives Sectors View Tools Help
Next Partition
Go

Boot Indicator: NO_SYSTEM

System ID: FAT FT, NTFS FT

Partition table index: Partition 1, Partition 2

Starting Head: 1

Ending Head: 15

Starting Sector: 1

Ending Sector: 63

Starting Cylinder: 813

Ending Cylinder: 832

Relative Sector: 63

Total Sectors: 20097

Tip:
Double click on the list box selections to change their values

For Help, press F1 | Drv 00 | Sec 819504 | Off 0x000

When the **System ID** field contains any value except EXTENDED or UNKNOWN, clicking the **Go** button next to the **Relative Sector** field reads the Partition Boot Sector for the logical drive.

To view the information for Partition 2 (the next logical volume), double-click Partition 2 or click **Next Partition** while viewing the information for Partition 1.

The **Relative Sector** field for each Partition 2 entry in an extended partition is the offset from the beginning of the extended partition, not the offset from the beginning of the Partition Table associated with the Partition 1 entry. Therefore, do not use the **Go** button to try to read the Partition Boot Sector for Partition 2.

Note Sometimes, the System ID byte for NTFS volume sets, stripe sets, mirror sets, and stripe sets with parity does not have the correct value. It should be 0x87, which DiskProbe displays as NTFS FT. Instead, it might be 0x86, which DiskProbe displays as FAT FT. Windows NT uses information in the Partition Boot Sector to determine which file system to use, so this incorrect value causes no problems, although it can be confusing to a person trying to understand the data on the disk. To be certain which file system is being used for a volume, look at the Partition Boot Sector.

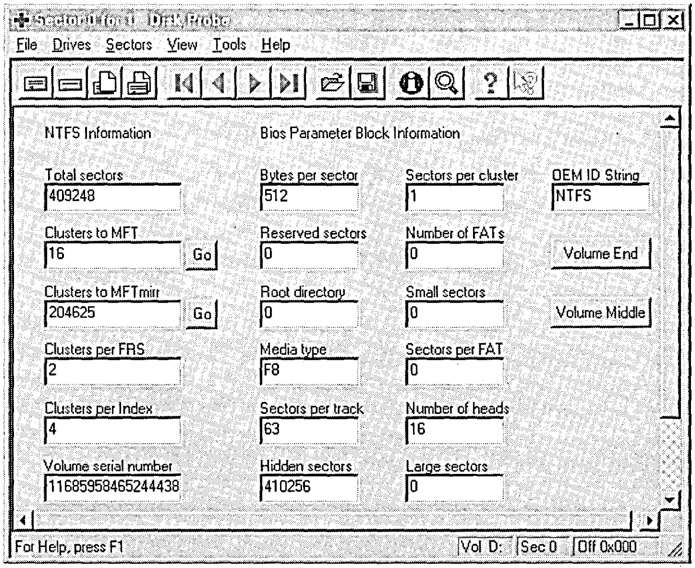
Viewing the Partition Boot Sector

You can use DiskProbe to view the information in Partition Boot Sectors. You see a formatted view of the information in the BIOS Parameter Block and the Extended BIOS Parameter Block. For a description of these structures, see "Partition Boot Sector," within Chapter 3, "Disk Management Basics."

▷ To view a Partition Boot Sector

1. In the Resource Kit folder, double-click the DiskProbe icon.
2. On the **Drives** menu, click **Logical Volume**. In the **Logical Volumes** list box of the **Open Logical Volume** dialog box, double-click the drive letter for the partition or logical drive whose Partition Boot Sector you want to read.
3. In the **Handle 0** group box, click **Set Active**. Click **OK**.
4. On the **Sectors** menu, click **Read** to display the **Read Sectors** dialog box. Set **Starting Sector** to 0 and **Number of Sectors** to 1.
5. On the **View** menu, click **NTFS Bootsector** or **FAT Bootsector** to see the information.

The next screen shot shows the NTFS Bootsector view of primary partition D for the disk configuration shown in Figure 7.1.



NTFS Information

Total sectors: 409248

Clusters to MFT: 16

Clusters to MFTmirr: 204625

Clusters per FRS: 2

Clusters per Index: 4

Volume serial number: 11685958465244438

Bios Parameter Block Information

Bytes per sector: 512

Reserved sectors: 0

Root directory: 0

Media type: F8

Sectors per track: 63

Hidden sectors: 410256

Sectors per cluster: 1

Number of FATs: 0

Small sectors: 0

Sectors per FAT: 0

Number of heads: 16

Large sectors: 0

OEM ID String: NTFS

Vol D: Sec 0 Off 0x000

When viewing an NTFS Partition Boot Sector, there are **Go** buttons next to the fields **Clusters to MFT** and **Clusters to MFT mirr**. When you click one of these buttons, it reads the first sector of the MFT or the MFT mirror, respectively. For information about these structures, see "Master File Table (MFT) and NTFS System Files" within Chapter 17, "Disk and File System Basics," in the *Windows NT Workstation Resource Guide.*

As described earlier in this section, Windows NT writes a backup of the NTFS Partition Boot Sector at the logical center of the volume (Windows NT version 3.51 and earlier) or the end of the volume (Windows NT 4.0). You can use the **Volume Middle** and **Volume End** buttons to read the backup Partition Boot Sector. However, these are some situations (when using Windows NT version 3.51 or earlier) where the backup Partition Boot Sector does not get written in the correct location:

- Converting a volume from FAT to NTFS.
- Extending a volume set.

View the sector you read to make sure it looks like an NTFS Partition Boot Sector, and that the information in it is accurate for the volume.

When viewing a FAT Partition Boot Sector, there is also a **Volume End** button, but there is no backup of the FAT Partition Boot Sector on the volume. If you are trying to find where volumes begin and end so you can repair a Partition Table, you can use the **Volume End** button for both FAT and NTFS partitions or logical drives to read what should be the sector immediately before the beginning of the next partition or logical drive.

Note If you are viewing a backup NTFS Partition Boot Sector, do not use the **Go**, **Volume End**, or **Volume Middle** buttons. These buttons read the correct sectors only when you are viewing the Partition Boot Sector at the beginning of the volume.

Backing Up and Restoring the Partition Boot Sector

The Partition Boot Sector contains information that the file system uses to access the volume. On x86-based computers, the Master Boot Record uses the Partition Boot Sector on the system partition to load the operating system kernel files, or in the case of Windows NT, the boot loader.

For examples of a Partition Boot Sector for a FAT and an NTFS volume, see Chapter 3, "Disk Management Basics."

There are several procedures that you can use, depending on whether you are backing up or restoring the Partition Boot Sector for:

- A primary partition
- A logical drive, which is a single contiguous area in an extended partition
- Several Partition Boot Sectors in an extended partition
- A volume set
- A stripe set
- A mirror set
- A stripe set with parity

You can open the disk to read the Partition Boot Sector by using either **Physical Drive** or **Logical Volume** on the **Drives** menu. When you use **Physical Drive**, you have to read the Partition Table to find the address of the Partition Boot Sector. When you use **Logical Volume**, you read the Partition Boot Sector by reading sector 0 of the logical drive. You cannot read the Master Boot Record when you use **Logical Volume**.

▷ **To back up a Partition Boot Sector for a primary partition by using Physical Drive**

1. Read the Partition Table and view it by using the procedure in "Displaying and Using the Partition Table," presented earlier in this section.
2. In the **Partition table index** list box, double-click the partition number for the Partition Boot Sector that you want to save. For instance, to save the Partition Boot Sector for the system partition, double-click the partition number that has the **Boot Indicator** set to **SYSTEM**.
3. To read the Partition Boot Sector, click the **Go** button next to the **Relative Sector** field. On the **View** menu, click **NTFS BootSector** or **FAT BootSector** to see the information displayed appropriately. Make sure that the sector you just read looks like a Partition Boot Sector, as described within Chapter 3, "Disk Management Basics."
4. On the **File** menu, click **Save As**. Enter the filename. Consider saving the file to the Emergency Repair Disk, the Windows NT startup floppy disk, or the MS-DOS bootable floppy disk. You can save it to more than one disk. When you are saving a Partition Boot Sector, using a name such as PBS*ssssssss*DISK*n*.dsk, where *ssssssss* is the relative sector of the Partition Boot Sector and *n* is the disk number, can help you remember what the file contains.

▶ **To back up the Partition Boot Sector by using Logical Volume**

1. In the Resource Kit folder, double-click the DiskProbe icon.
2. On the **Drives** menu, click **Logical Volume**. In the **Logical Volumes** list box, double-click the drive letter for the partition whose Partition Boot Sector you want to save.
3. In the **Handle 0** group box, click **Set Active**. Close the dialog box.
4. On the **Sectors** menu, click **Read**. Set **Starting Sector** to 0 and **Number of Sectors** to 1. Make sure that the sector you read looks like a Partition Boot Sector, as described within Chapter 3, "Disk Management Basics."
5. On the **File** menu, click **Save As**. Enter the filename. Consider saving the file to the Emergency Repair Disk, the Windows NT startup floppy disk, or the MS-DOS bootable floppy disk. You can save it to more than one disk. When you are saving a Partition Boot Sector in this manner, using a name such as PBS*d*DSK*n*.dsk, where *d* is the drive letter of the logical drive, and *n* is the disk number, can help you remember what the file contains. Using a name such as this assumes that you will not be changing drive letters.

▶ **To back up the Partition Boot Sectors in an extended partition by using Physical Drive**

1. Read the Partition Table and view it by using the procedure in "Displaying and Using the Partition Table," presented earlier in this section.
2. In the **Partition table index** list box, double-click the partition number for the extended partition, and then click **Go**. Doing this reads the first Partition Table entry in the extended partition.
3. You can walk the extended partition to find the logical drive whose Partition Boot Sector you want to back up. When you walk the extended partition, view the information for Partition 1 and click **Go** to read the Partition Boot Sector. You can use the **Search** function on the **Tools** menu to find each Partition Boot Sector. See "Finding a Partition Boot Sector," presented later in this chapter, for information about finding these disk sectors.

   You can also print a map of the disk by using the DiskMap utility, which provides you with the address of the Partition Table entry for each logical drive. You can then use DiskProbe to read the first sector of the logical drive whose Partition Boot Sector you want to back up, and use the **Go** button to read the Partition Boot Sector.

   Once you have read a Partition Boot Sector, on the **View** menu, click **NTFS BootSector** or **FAT BootSector** to see the information displayed appropriately. Make sure that the sector you just read looks like a Partition Boot Sector, as described within Chapter 3, "Disk Management Basics."

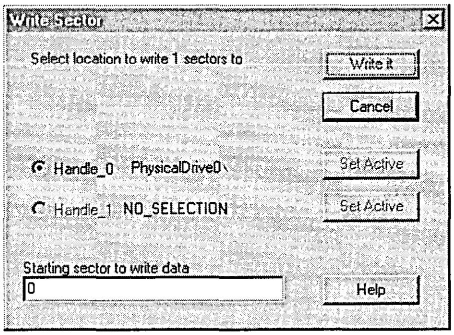
4. On the **File** menu, click **Save As**. Enter the filename. Consider saving the file to the Emergency Repair Disk, the Windows NT startup floppy disk, or the MS-DOS bootable floppy disk. When you are saving a Partition Boot Sector, using a name such as PBS*ssssssss*DISK*n*.dsk, where *ssssssss* is the relative sector of the Partition Boot Sector and *n* is the disk number, can help you remember what all of the files contain.

▷ **To restore the Partition Boot Sector for a primary partition by using Physical Drive**

1. In the Resource Kit folder, double-click the DiskProbe icon.
2. On the **Drives** menu, click **Physical Drive**. In the **Available Physical Drives** group box, the disks are listed as PhysicalDrive*n*, where n=0 for the first hard disk. Double-click the disk that contains the Partition Boot Sector that you want to replace.
3. In the **Handle 0** group box, clear the **Read Only** check box, click **Set Active,** and click **Close.**
4. If you know the address of the sector you need to write, skip to step 6. To use DiskProbe to compute the sector, first read the Partition Table and view it by using the procedure in "Displaying and Using the Partition Table," presented earlier in this section. Figure 7.3 shows the DiskProbe Partition Table view.
5. In the **Partition table index** list box, double-click the partition number for the Partition Boot Sector that you want to replace. For example, to view information about the Partition Boot Sector for the system partition, double-click the partition number that has the **Boot Indicator** set to **SYSTEM.**

   The **Relative Sector** field contains the sector number for the Partition Boot Sector. Write down this number.
6. On the **File** menu, click **Open**, and enter the filename of the file that contains the Partition Boot Sector. Open the file. Verify that it looks like a Partition Boot Sector.

7. On the **Sectors** menu, click **Write** to display the **Write Sector** dialog box. In the **Starting sector to write data**, enter the sector number from step 5 (or that you already know is the correct sector). Click the option button for the disk to which you want to write the sector. Click **Write it.**



Select location to write 1 sectors to

● Handle_0 PhysicalDrive0\

○ Handle_1 NO_SELECTION

Starting sector to write data: 0

8. DiskProbe displays a message box so you can verify the information.

▷ **To restore the Partition Boot Sector by using Logical Volume**

1. In the Resource Kit folder, double-click the DiskProbe icon.
2. On the **Drives** menu, click **Logical Volume**. In the **Logical Volumes** list box of the **Open Logical Volume** dialog box, double-click the logical drive whose Partition Boot Sector you want to restore.
3. In the **Handle 0** group box, clear **Read Only,** click **Set Active**, and click **Close.**
4. On the **File** menu, click **Open**, and enter the filename of the file that contains the Partition Boot Sector. Open the file. Verify that it looks like a Partition Boot Sector and is 512 bytes. DiskProbe displays an error if the file size is not a multiple of 512.
5. On the **Sectors** menu, click **Write** to display the **Write Sector** dialog box. In the **Starting sector to write data**, enter 0 for the **Starting sector to write data.** Click **Write it.**
6. DiskProbe displays a message box so you can verify the information.

▷ To restore the Partition Boot Sector in an extended partition by using Physical Drive

1. If you know the address of the Partition Boot Sector that you want to restore (which you would know if the sector number is part of the filename), follow the procedure for restoring the Partition Boot Sector (given earlier in this chapter) for a primary partition, skipping steps 4 and 5. You are done.

   If you do not know the address, you can find it by walking the extended partition. Follow steps 1 through 3 in the procedure for backing up the Partition Boot Sector for an extended partition (given earlier in this chapter). Even though the Partition Boot Sector that you read is the one you want to replace, when you read it, DiskProbe displays its relative sector number. Write down this number.

   You can also calculate the address of the Partition Boot Sector by using a DiskMap output for the disk. The sector number for a Partition Boot Sector is the address of the EBR for the logical drive + the Relative Sector value.
2. In the **Handle 0** group box of the **Open Physical Drive** dialog box, be sure that **Read Only** is cleared.
3. On the **File** menu, click **Open**, and enter the filename of the file that contains the Partition Boot Sector. Open the file. Verify that it looks like a Partition Boot Sector.
4. On the **Sectors** menu, click **Write** to display the **Write Sector** dialog box. In the **Starting sector to write data**, enter the sector number that you found in step 1. Click the option button for the disk to which you want to write the sector. Click **Write it**.
5. DiskProbe displays a message box so you can verify the information.

If you formatted an NTFS volume when running Windows NT version 4.0, a backup of the Partition Boot Sector for NTFS primary partitions and logical drives is located at the end of the partition or logical drive. Earlier versions of Windows NT put the backup at the center of the volume. If you did not make a backup of the Partition Boot Sector, you can find the backup sector and copy it to the first sector of the volume.

▷ To restore the backup NTFS Partition Boot Sector

1. In the Resource Kit folder, double-click the DiskProbe icon.
2. “Finding a Partition Boot Sector,” presented later in this chapter, describes different methods for finding the Partition Boot Sector. When you find a Partition Boot Sector within the boundaries of an NTFS volume, make sure that the data are reasonable for the volume. If you have a deleted volume, or extended an NTFS volume, there might be old, invalid, Partition Boot Sectors on the disk. Save the backup to a disk.
3. Close any handles that you opened to find the backup.

4. On the **Drives** menu, click **Logical Volume**. Click the **Close Handle** button for any drives or logical volumes that are currently open. In the **Logical Volumes** list box of the **Open Logical Volume** dialog box, double-click the logical drive whose Partition Boot Sector you want to restore.
5. In the **Handle 0** group box, clear **Read Only** and click **Set Active**. Click **OK**.
6. On the **File** menu, click **Open**, and enter the name of the file you copied to disk in step 2. Open the file.
7. On the **Sectors** menu, click **Write** to display the **Write Sector** dialog box. Since you are going to be writing to relative sector 0, make sure that the active handle is a drive letter, not PhysicalDrive*n*. Enter 0 for the **Starting sector to write data**. Click **Write it**.
8. DiskProbe displays a message box so you can verify the information.

Finding a Partition Boot Sector

This section discusses some approaches for finding primary partitions and logical drives on your disk in order to repair the Partition Table, which is discussed in "Editing and Repairing the Partition Table," presented later in this chapter.

When you are looking for Partition Boot Sectors in order to repair the Partition Table, you know that some information is wrong or missing. Therefore, you need to verify that the information you use to find a Partition Boot Sector, or a Partition Table in an extended partition, is accurate. For example, if you use the **Number of Sectors** field for one primary partition to calculate the location of the Partition Boot Sector for the next partition, make sure that there is a valid Partition Boot Sector at that sector.

The easiest way to find the start of a primary partition is to find the Partition Boot Sector for the partition. For an extended partition, subtract the sectors per track from the address of the Partition Boot Sector to find the Partition Table entry for the logical drive.

The Partition Boot Sector for the first partition on your hard disk is always the first sector on head 1 of cylinder 0. Because disks typically have 32 or 63 sectors per track, you can usually find the Partition Boot Sector for the first partition at either relative sector 32 or relative sector 63. You can get the sectors-per-track information by clicking **Volume Information** on the **Drives** menu.

If you think that the relative sector field for a partition or logical drive is correct, you can read the sector by clicking **Go** when you are viewing its information to see if it looks like a Partition Boot Sector.

There is also a **Total Sectors** field on the Partition Table view. If you think this field is accurate, you can use this value to calculate the start of the next primary partition or logical drive.

The FAT and NTFS Partition Boot Sectors each have a field that contains the total number of sectors. If you already know or have found the start of a partition or logical drive, you can add the number of sectors to the starting sector to find the start of the next partition or logical drive. For an NTFS Partition Boot Sector, on the **View** menu, click **NTFS BootSector**. The **Total sectors** field contains the value you want. For a FAT volume, on the **View** menu, click **FAT Bootsector**. If the size of the primary partition or logical drive fits in 16 bits, the **Small sectors** field contains the size. Otherwise, the size is in the **Large sectors** field.

You can also use DiskProbe to search for Partition Boot Sectors. As you find each one, write down its location, and use the Partition Table view in DiskProbe to enter the information.

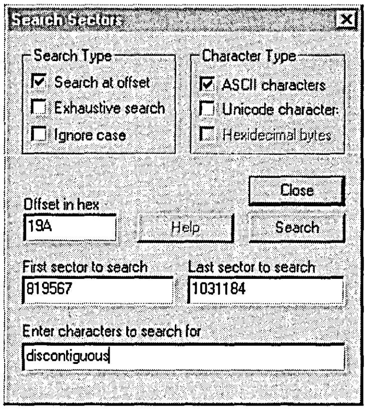
There are three situations in which you can find a Partition Boot Sector that is not at the beginning of a volume. You need to know about these other Partition Boot Sectors so you do not use the information about where these Partition Boot Sectors are located to repair the Partition Table.

- If you formatted an NTFS volume, or converted a volume from FAT to NTFS when running Windows NT version 4.0, each NTFS volume has a backup of its Partition Boot Sector at the end of the volume. Earlier versions of Windows NT put the backup at the middle of the volume when you formatted it, but did not create a backup when you converted the volume.
- When you extend an NTFS volume set, Windows NT version 4.0 puts a new copy of the Partition Boot Sector in the middle of the extended volume set. It does not remove the old copy of the Partition Boot Sector. If you extend a volume set when running Windows NT version 3.51, it does not remove the old copy of the Partition Boot Sector either, nor does it write a new one.
- There can be old, invalid Partition Boot Sectors on the disk if you have deleted volumes.

▷ **To search for Partition Boot Sectors**

1. In the Resource Kit folder, double-click the DiskProbe icon.
2. On the **Drives** menu, click **Physical Drive**. The **Available Physical Drives** in the **Open Physical Drive** dialog box are listed as PhysicalDrive*n*, where *n*=0 for the first hard disk. Double-click the disk that you want to open, click **Set Active**, and click **Close**.
3. On the **Tools** menu, click **Search**. The information that you search for depends upon whether you want to find a FAT or an NTFS Partition Boot Sector. If you check the **Search at offset** check box and specify an **Offset in hex** at which to search for a specific value, the search runs much faster than searching each entire sector. You can search the entire disk if you do not change the default values in the **First sector to search** and **Last sector to search** fields.

This is an example of a search for an NTFS Partition Boot Sector.



Search Sectors

Search Type
☑ Search at offset
□ Exhaustive search
□ Ignore case

Character Type
☑ ASCII characters
□ Unicode character
□ Hexidecimal bytes

Offset in hex: 19A

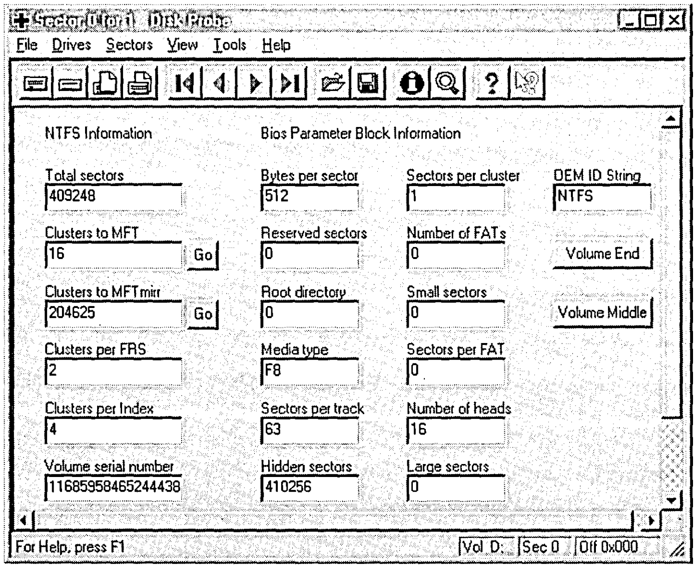
First sector to search: 819567

Last sector to search: 1031184

Enter characters to search for: discontiguous

Note The contents of the Partition Boot Sector for an NTFS volume can be slightly different, depending on whether you formatted the volume when running Windows NT version 4.0 or earlier versions of Windows NT. If you specify a value in the **Offset in hex** field, you might need to search at a different offset depending on the data you are searching for, and which version of Windows NT you used to format the volume. For example, the string "discontinguous" starts at location 19A when you use Windows NT version 4.0 to format the volume. But discontinguous is at hex location 195 if you formatted the volume when running Windows NT version 3.5. The string NTFS starts at offset 3, regardless of the version of Windows NT you used to format the volume. If you don't know which version of Windows NT was used to format a volume, search for NTFS. Note that DiskProbe will find matches in sectors that are not Partition Boot Sectors.

4. If DiskProbe finds a match, it displays the address of the sector, as the following figure shows:



NTFS Information

Total sectors
409248

Clusters to MFT
16

Clusters to MFTmirr
204625

Clusters per FRS
2

Clusters per Index
4

Volume serial number
11685958465244438

Bios Parameter Block Information

Bytes per sector
512

Reserved sectors
0

Root directory
0

Media type
F8

Sectors per track
63

Hidden sectors
410256

Sectors per cluster
1

Number of FATs
0

Small sectors
0

Sectors per FAT
0

Number of heads
16

Large sectors
0

OEM ID String
NTFS

Be sure to verify that each sector DiskProbe finds is really a Partition Boot Sector. The **View** menu enables you to look at the data as bytes or formatted as an NTFS or a FAT Partition Boot Sector.

Editing and Repairing the Partition Table

You can use DiskProbe to display and change your Partition Table. Repairing a damaged Partition Table is risky, and should never be attempted by a user not familiar with the procedure. If all of the partitions on the disk are primary partitions, the repair process is much simpler than if you have an extended partition, volume sets, stripe sets, mirror sets, or stripe sets with parity.

If the wrong partition is set as the system partition, or you do not have a system partition on the disk, you can use DiskProbe to change this information in the Partition Table. If you have configured your system partition as a mirror set, you need to set the Boot Indicator field on the shadow partition. If this flag is not set and the original partition or the disk with the original partition fails, you will have to use the Windows NT startup floppy disk to start Windows NT.

Note The system partition is always a primary partition. The original partition is the one from which you created the mirror set. The copy of the original partition is called the shadow partition.

▷ **To change or set the Boot Indicator field**

1. View the Partition Table by using the procedure described in "Displaying and Using the Partition Table," presented earlier in this chapter.
2. Double-click each partition number in the **Partition table index** list box to determine which one is the system partition, or that none is set as the system partition. The partition that has the **Boot Indicator** field set to SYSTEM is the system partition.
3. To clear the **Boot Indicator** field for a partition, double-click its partition number, double-click **NOSYSTEM** in the **Boot Indicator** list box.
4. To set the system partition, double-click its partition number, and double-click **SYSTEM** in the **Boot Indicator** list box.
5. Make sure that the disk is in read/write mode. On the **Drives** menu, click **Physical Drive** to see the **Open Physical Drive** dialog box. Clear the **Read Only** check box, if it is checked. Click **Close.**
6. On the **Sectors** menu, click **Write**, and enter 0 in **Starting sector to write to.** Click **Write it.**
7. DiskProbe displays a message box so you can verify the information.

When you create a volume set, stripe set, mirror set, or stripe set with parity, Windows NT sets the fault tolerant (FT) bit of the System ID byte in the Partition Table. This bit indicates that Windows NT needs to use the Registry key HKEY_LOCAL_MACHINE\SYSTEM\DISK to know how to find all of the members of the volume. Because volumes cannot be accessed without the DISK key, they can only be accessed by Windows NT. In a dual-boot configuration, the other operating system cannot use a partition with the FT bit set.

In general, you should use Disk Administrator, and not DiskProbe, to change the value of this field. You should only change this field to remove the FT flag, and then only to break a mirror set. You might want to set or clear the FT flag to test recovery procedure, but be sure to have a backup of the Registry and the data on the partition before you do so.

▷ To clear the FT flag for a primary partition

1. View the Partition Table by using the procedure described in "Displaying and Using the Partition Table," presented earlier in this chapter.
2. Double-click the partition number in the **Partition table index** list box for which you want to change the value of the FT flag.
3. If you want to remove the FT flag from a FAT partition, you need to figure out whether the partition should be FAT 12 bit, FAT 16 bit, or FAT Large. See the section titled "System ID Field," within Chapter 3, "Disk Management Basics," for information about the types of FAT partitions. Double-click either **FAT 12 bit**, **FAT 16 bit**, or **FAT Large** in the System ID field.
4. If you want to remove the FT flag from a NTFS partition, double-click **NTFS** in the System ID field.
5. Make sure that the disk is in read/write mode. On the **Drives** menu, click **Physical Drive** to see the Open Physical drives dialog box. Clear the **Read Only** check box, if it is checked. Click **Close**.
6. On the **Sectors** menu, click **Write**, and enter 0 in **Starting sector to write to**. Click **Write it**.
7. DiskProbe displays a message box so you can verify the information.

When you have found the Partition Boot Sectors, you can calculate all of the information for the Partition Table entries. If you do not know the values for the number of heads, number of sectors per track, or number of cylinders for the disk, open a handle by clicking either **Physical Drive** or **Logical Volume** on the **Drives** menu, and then click **Volume Information** on the **Drives** menu.

If none of the information in your Partition Table is correct, you might be safer to try to do a complete backup of the data on the disk. It might even be faster to backup and restore the data instead of repairing the Partition Table. You can then recreate and reformat the partitions and logical drives, and restore all of the data.

Note If a partition or logical drive extends beyond cylinder 1023, set the Starting and Ending Sector, Head, and Cylinder fields to the maximum values. Maximum values indicate to Windows NT that it needs to use the information in the Partition Boot Sector to know how to access the volume. Remember that your system and boot partitions should not go beyond cylinder 1023 if you have set up your computer to dual-boot MS-DOS or Windows 95.

▷ **Repairing the Partition Table**

1. For primary partitions, view the Partition Table by using the procedure described in "Displaying and Using the Partition Table," presented earlier in this chapter. For an extended partition, the section "Walking an Extended Partition," presented earlier in this chapter, describes finding the Partition Table entry for each logical drive.
2. Double-click the partition number for which you want to change information. Enter the new values.
3. Make sure that the disk is in read/write mode. On the **Drives** menu, click **Physical Drive** to see the **Open Physical Drive** dialog box. Clear the **Read Only** check box, if it is checked. Click **Close**.
4. On the **Sectors** menu, click **Write**, and enter the sector number of the Partition Table that you read in step 1. If you are changing information for primary partitions, the sector number is 0. Click **Write it**.
5. DiskProbe displays a message box so you can verify the information.

DiskSave — Back Up and Restore Critical Disk Sectors

DiskSave is an MS-DOS-based utility that enables you to save the Master Boot Record (MBR) and Partition Boot Sector (PBS) as binary files. You must start MS-DOS to run this utility. It will not run from the command prompt.

You should save these files to your Windows NT startup floppy disk or your MS-DOS bootable floppy disk. If your computer fails to start because it has a problem using the MBR or PBS, you can restore them and try the startup again.

This tool also enables you to turn off the FT bit in the System ID field of the system partition. You should turn off the bit on the system partition on the shadow disk if you are unable to start Windows NT when the original disk has failed.

The Master Boot Record contains code that the system BIOS on x86-based computers uses to read the Partition Table and find the Partition Boot Sector of the system partition. This sector also contains the Partition Table. If this sector becomes damaged, the computer cannot find or start the operating system.

The Partition Boot Sector contains code that loads the operating system kernel or a boot loader. A corrupt Partition Boot Sector can also cause startup failures.

Note If either of the sectors that you restore does not match the configuration of your computer, you can make your problems worse. Always be sure to save the Master Boot Record and the Partition Boot Sector whenever you make changes that affect them.

These are the DiskSave functions:

- F2, back up the Master Boot Record. This function saves cylinder 0, head 0, sector 1 of the startup disk to the filename that you enter. You need to save a new copy of this record any time that you create or delete a partition, or change the file system used on a partition.
- F3, restore the Master Boot Record. This function copies the file that you specify to cylinder 0, head 0, sector 1 of the startup disk. Restoring this sector also replaces the Partition Table. There are no checks to determine if the sector is a valid Master Boot Record.
- F4, backup the Partition Boot Sector. This function saves the first sector of the system partition on disk 0 to the filename specified.
- F5, restore the Partition Boot Sector. This function replaces the first sector of the system partition with the contents of the specified file. There are no checks to determine if the sector is a valid Partition Boot Sector.
- F6, disable fault tolerance on the startup disk. This function is useful when Windows NT will not start from a mirror set of the system partition. The function looks for the system partition. It then checks to see if the System ID byte has the high bit set. Windows NT sets the high bit of the System ID field if the partition is a member of a fault-tolerant volume. Disabling this bit has the same effect as breaking the mirror set. DiskSave does not allow you to reenable the bit once it has been disabled.

  If you use F6 to turn off the FT bit to be able to start, you still need to break the mirror set after you have started Windows NT. Use Disk Administrator to break the mirror set, which turns off the FT bit in the shadow partition. You need to do this so the Registry and disk information match.

When DiskSave makes calls to the system BIOS, it checks the status of each call. The normal return value is zero. Originally, a nonzero value indicated an error, and the error codes were documented. Currently, there are system BIOSs that return nonzero values, which are not documented. These nonzero values do not seem to be errors. So far, all of these nonzero values that are not errors have occurred when using IDE disks.

If the value is non-zero, DiskSave displays a message at the bottom of the screen. If you see a message about a system BIOS call returning a nonzero value, it would be a good idea to check whether the function worked correctly. One way to check is to start Windows NT, and use the Windows NT-based utility DiskProbe to compare the sector that you tried to save or restore with a copy of the data. For example, if you see a message about a nonzero return code when you select F4 to save the Partition Boot Sector, finish saving the sector to a file. Then, using DiskProbe, read the Partition Boot Sector from disk, and compare the contents to the file that you just saved.

▷ To use DiskSave to back up the Master Boot Record or Partition Boot Sector

1. Start MS-DOS. Enter **disksave** at the command prompt. You now have the following options:

   F2 — Backup the Master Boot Record

   F3 — Restore Master Boot Record

   F4 — Backup the Boot Sector

   F5 — Restore Boot Sector

   F6 — Disable FT on the Boot Drive

2. Press F2 to save the Master Boot Record and F4 to save the Partition Boot Sector. You need to enter the full name for the file, including the path. For example, to save the Master Boot Record to a folder on your C drive, use a name such as C:\Backup\Mbrdisk0.dsk. If you copy it to a floppy disk, the path should be similar to A:\Mbrdisk0.dsk.

Note DiskSave only saves the Master Boot Record on disk 0 and the Partition Boot Sector for the system partition on disk 0.

▷ **To use DiskSave to replace the Master Boot Record or Partition Boot Sector**

1. Start MS-DOS. Enter **disksave** at the command prompt. You have the following options:

   F2 — Backup the Master Boot Record

   F3 — Restore Master Boot Record

   F4 — Backup the Boot Sector

   F5 — Restore Boot Sector

   F6 — Disable FT on the Boot Drive

2. Press F3 to replace the Master Boot Record and F5 to replace the Partition Boot Sector. You need to enter the full name for the file, including the path. For example, if you have saved the Master Boot Record on your Emergency Repair Disk or the Windows NT startup floppy disk, the path might be A:\Mbrdsk0.dsk.

Note DiskSave only restores the Master Boot Record on disk 0 and the Partition Boot Sector for the system partition on disk 0.

FTEdit — Recovering Volume Sets, Stripe Sets, and Stripe Sets With Parity

The FTEdit utility on the *Windows NT Server Resource Kit* CD aids in the recovery of stripe sets, volume sets, stripe sets with parity, and mirror sets. For more information about using FTEdit, see Ftedit.hlp on the *Windows NT Server Resource Kit* CD.

Note You can recover and rebuild information for all of these volumes in the same way. However, rebuilding a mirror set with FTEdit is a waste of time, since the mirror set has to be regenerated anyway. You are better off simply breaking the mirror set by using Disk Administrator, and then recreating it. See Disk Administrator Help for the procedures.

Because stripe sets and volume sets are not fault tolerant, Windows NT Server performs orphan processing differently than it would for stripe sets with parity and mirror sets. If one of the disks in a stripe set or volume set fails, the entire volume is no longer usable, and all the primary partitions or logical drives that are members of the volume are marked as orphans. In Disk Administrator, the members of the volume are displayed as Unknown.

All data about the configuration of volume sets, stripe sets, mirror sets, and stripe sets with parity are contained in the HKEY_LOCAL_MACHINE\SYSTEM\DISK Registry key. This disk configuration information is stored in binary format, so it is not practical to edit it manually. When the Registry information is corrupt or missing, and no backups are available, you can use FTEdit to build the Registry information and allow the operating system to read the volumes.

For example, if the disk containing the boot partition becomes unusable, or the computer fails, you might want to move the disks containing the volumes to another computer. Alternatively, you might decide to reinstall the operating system on a new disk. In either case, the operating system does not have any information about the volumes.

The operating system has information about which disks are members of volumes, but it cannot distinguish between a stripe set, a stripe set with parity, a volume set, or a mirror set without the Registry information. In this situation, when the volumes are displayed in Disk Administrator, the file system might be displayed as Unknown, and there will be no drive letter assigned to the volumes. This is to prevent writing to the volume, which could corrupt it.

When using FTEdit to rebuild the DISK Registry subkey, you need to know the order in which the partitions on the disks were combined to create the volume. The following scenario shows why order is important.

A previous user created a volume set by combining unpartitioned areas on three disks in this order:

- 2 MB on disk 2
- 3 MB on disk 0
- 4 MB on disk 1

When a volume set is created, the Partition Boot Sector is the first sector on the first area selected. Data are written to the areas in the order in which they are combined. Thus, if someone created a 6 MB file on this volume set, the file would occupy the following areas:

- 2 MB on disk 2
- 3 MB on disk 0
- 1 MB on disk 1

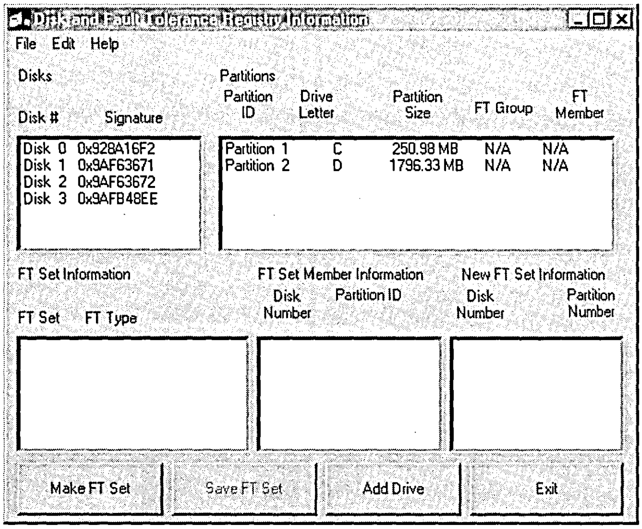
Now you need to use FTEdit to rebuild the volume set. You select the disks in the order disk 0, disk 1, and then disk 2. Windows NT will not be able to find the Partition Boot Sector for the volume set, because the first sector of the volume set that you rebuilt is on disk 0, which contains only data. The Partition Boot Sector is actually on disk 2, but that disk area is at the end of the volume set you created.

Your attempt to rebuild the volume set has failed, because you did not know the order in which to combine the disk areas.

The remainder of this section describes using FTEdit to build Registry information for a stripe set with parity that has been moved from another computer. (Disks 1, 2, and 3 are the disks that have been moved.) Each disk is 519 MB and has one partition. Both computers are running Windows NT Server version 4.0.

▷ To rebuild a stripe set with parity

1. Open Disk Administrator, so that you can update the disk configuration information. Disk Administrator needs to store the disk signature and other basic disk information before the disks can be recognized.
2. Open FTEdit. The dialog box in Figure 7.4 is displayed.



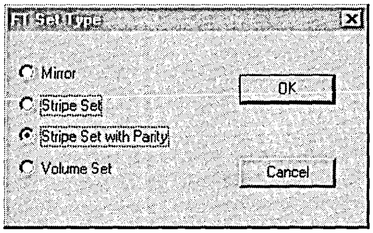
Disk #	Signature
Disk 0	0x928A16F2
Disk 1	0x9AF63671
Disk 2	0x9AF63672
Disk 3	0x9AFB48EE

Partition ID	Drive Letter	Partition Size	FT Group	FT Member
Partition 1	C	250.98 MB	N/A	N/A
Partition 2	D	1796.33 MB	N/A	N/A

Figure 7.4 Using FTEdit to recover a stripe set with parity

The **Disks** group box identifies all of the disks connected to the computer. The **Partitions** group box contains information about the disk currently selected, or the first disk, if none are selected. Disk 0 is two gigabytes, and contains two partitions.

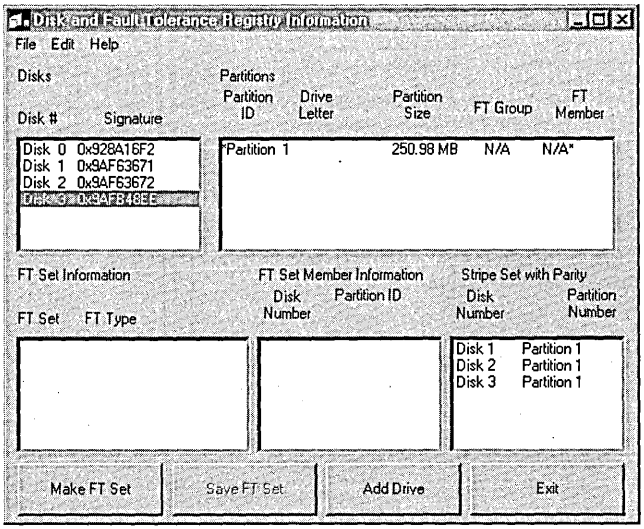
3. On the **Edit** menu, click **Create FT Set**. In the **FT Set Type** dialog box, click the Option button for the type of volume you are building. In this example, click **Stripe Set with Parity**.



○ Mirror
○ Stripe Set
● Stripe Set with Parity
○ Volume Set

4. In the **Disks** list box, select the first disk that will be a member of the stripe set with parity. Available partitions are displayed in the **Partitions** list box on the right.

   Double-click the partition from each disk that will make up the stripe set with parity. The disk and partition information will be displayed in the list box on the lower right. The title of this list box reflects the type of volume being built. For this example, the title is **Stripe Set with Parity**. The next screen shot shows the dialog box after double-clicking Partition 1 on each of Disk 1, Disk 2, and Disk 3.



Disks

Disk #	Signature
Disk 0	0x928A16F2
Disk 1	0x9AF63671
Disk 2	0x9AF63672
Disk 3	0x9AFB48EE

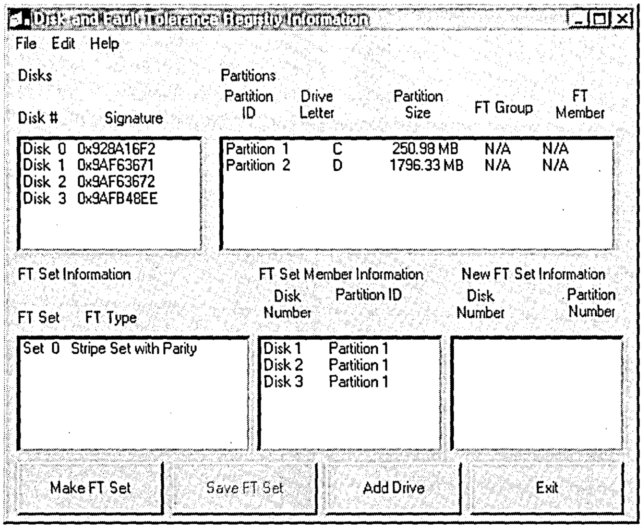
Partitions

Partition ID	Drive Letter	Partition Size	FT Group	FT Member
*Partition 1		250.98 MB	N/A	N/A*

Stripe Set with Parity

Disk Number	Partition Number
Disk 1	Partition 1
Disk 2	Partition 1
Disk 3	Partition 1

5. Click the **Save FT Set** button. The information is transferred to the **FT Set Member Information** list box. Check to make sure the information is correct. FTEdit should look like the following illustration:



Disks

Disk #	Signature
Disk 0	0x928A16F2
Disk 1	0x9AF63671
Disk 2	0x9AF63672
Disk 3	0x9AFB48EE

Partitions

Partition ID	Drive Letter	Partition Size	FT Group	FT Member
Partition 1	C	250.98 MB	N/A	N/A
Partition 2	D	1796.33 MB	N/A	N/A

FT Set Information

FT Set	FT Type
Set 0	Stripe Set with Parity

FT Set Member Information

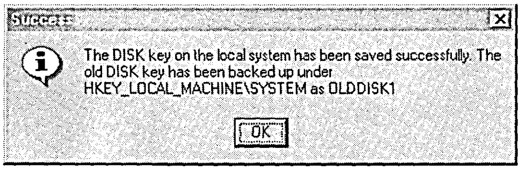
Disk Number	Partition ID
Disk 1	Partition 1
Disk 2	Partition 1
Disk 3	Partition 1

New FT Set Information

Disk Number	Partition Number

The **Partitions** list box shows the information for Disk 0. The disks that are used in creating the new volume are those listed in the **FT Set Member Information** list boxes.

6. On the **Edit** menu, click **Save Changes to System**. The following message box is displayed:



The DISK key on the local system has been saved successfully. The old DISK key has been backed up under HKEY_LOCAL_MACHINE\SYSTEM as OLDDISK1

7. Exit FTEdit, close any open programs, and restart the system. You need to do this to get the new information loaded into the Registry.

8. Open Disk Administrator.

   Click one of the partitions in the stripe set with parity. If the status bar has a message like **Stripe set with parity #N [INITIALIZING]**, the computer on which you were previously using the stripe set with parity did not finish shutdown. Wait for Disk Administrator to finish the initialization. If the volume was functional before the disks were moved, and the correct information was entered in FTEdit, the data on the volume should be intact.

9. From the Disk Administrator **Tools** menu, click **Drive Letter**. In the **Assign Drive Letter** dialog box, click **Assign drive letter**, and then choose a drive letter for the new volume.

   The volume is now accessible from Windows NT Explorer, My Computer, and by the rest of the operating system.

10. After confirming that the volume is correctly configured and is accessible, be sure to add this new configuration information to the records that you keep for this computer.

   You might want to back up the stripe set with parity at this time.

These same steps can be used to recover volume sets and stripe sets. To successfully use FTEdit to build Registry information, you must know the following information:

- Which disks contain stripe sets, stripe sets with parity, and volume sets.
- Which partitions belong to the stripe sets, stripe sets with parity, and volume sets.
- The order in which the partitions were combined into the stripe sets, stripe sets with parity, or volume sets.

If your configuration involves multiple stripe sets, stripe sets with parity, and volume sets on several different disks, you might face a very difficult job remembering which primary partitions and logical drives belong to which volumes. For this reason, it is a good idea to keep the disk configuration simple, and to keep recovery information as up-to-date as possible.

Note FTEdit edits the binary information in the Registry. It cannot recover damaged or corrupt data on the disk.

File System Utilities

The utilities described in this section are available on either the Windows NT Server product CD, or the *Windows NT Server Resource Kit* CD. This table shows which ones are on which CD.

Utility	Location
Cacls	product CD
Compact	product CD
Compress	resource kit CD
Convert	product CD
Diruse	resource kit CD
Expand	product CD
Expndw32	resource kit CD

Cacls: Changes ACLs of NTFS Files and Folders

You can use the **cacls** utility to display or modify access control lists (ACL) of files or folders. A description of the command options follows the syntax. This is the format of the command:

```
CACLS <filename|folder> [/t] [/e] [/c] [/g user:perm] [/r user [...]]
[/p user:perm [...]] [/d user [...]]
```

Option	Description
filename or *folder name*	Displays ACLs.
/t	Changes ACLs of specified files in the current folder and all subfolders.
/e	Edit ACL instead of replacing it.
/c	Continue on access denied errors.
/g user:perm	Grant specified user access rights, where perm can be: R Read C Change (write) F Full control
/r user	Revoke specified user's access rights (only valid with /e).
/p user:perm	Replace specified user's access rights, where perm can be: N None R Read C Change (write) F Full control
/d user	Deny access to the specified user.

Wildcards can be used to specify more than one file in a command. You can specify more than one user in a command also.

If you already have permissions set for multiple users on a folder or file and do not use the /e option, all user permissions are removed except for the user and permissions specified on the command line. You should use the following syntax when modifying user permissions to include read, change, and full control respectively:

```
cacls <filename|folder> /e /r <username>
cacls <filename|folder> /e /g <username>:<permission>
cacls <filename|folder> /e /p <username>:<permission>
```

The **cacls** utility does not provide a **/y** option that answers automatically with Y for Yes to the **ARE YOU SURE? Y/N** prompt. However, you can use the **echo** command to pipe the character Y as input to the prompt when you are running **cacls** in a batch file. Use the following syntax to automatically answer Y:

```
echo y| cacls <filename|folder> /g <username>:<permission>
```

Note Do not enter a space between the y and the pipe symbol (|). If you do, **cacls** fails to make the permission change.

Compact: Compresses and Uncompresses NTFS Files and Folders

The **compact** utility is the command line version of the compression functionality in My Computer or Windows NT Explorer. The **compact** command displays and alters the compression of folders and files on NTFS volumes. It also displays the compression state of folders. For more information about this utility, type **compact /?** at the command prompt. The following table describes the options.

The format of the command is:

```
compact [/c] [/u] [/s[:folder]] [/a] [/i] [/f] [/q] [filename [...]]
```

Option	Description
none	Displays the compression state of the current folder.
/c	Compresses the specified folder or file.
/u	Uncompresses the specified folder or file.
/s[*folder*]	Specifies that the requested action (compress or uncompress) be applied to all subfolders of the specified folder, or to the current folder if none is specified.
/i	Ignores errors.
/f	Forces compression or uncompression of the specified folder or file.
/a	Displays files with the hidden or system attribute.
/q	Reports only the most essential information.
filename	Specifies a pattern, file, or folder. You can use multiple filenames and wild cards.

There are reasons why you would want to use this utility instead of My Computer or Windows NT Explorer:

- You can use compact in a batch script. Using the **/i** option enables you to skip files that cannot be opened when you are running in batch mode, such as when a file is already in use by another program.
- If the system crashed when compression or uncompression was occurring, the file or folder is marked as Compressed or Uncompressed, even if the operation did not complete. You can force the operation to complete by using the **compact** utility with the **/f** option (with either the **/c** or **/u**).

Note The Compact utility automatically compresses or uncompresses all of the files and subfolders when you change the compression state of a folder. It does not ask whether you want to change the compression state of the files or subfolders within it.

Compress: Compresses Files or Folders

This command line utility can be used to compress one or more files. You cannot open a file that has been compressed by using this utility. You have to expand it by using Expand or Expndw32 first.

To use this utility, type **compress** with the appropriate options at the command line:

compress [**-r**] [**-d**] *source* [*destination*]

Option	Description
-r	Renames compressed files.
-d	Updates compressed files only if out of date.
source	Specifies the source file. The "*" and "?" wildcards can be used.
destination	Specifies the destination file or path. The destination can be a folder. If source specifies multiple files and the **-r** option is not specified, then *destination* must be a folder.

Note You should not use this utility to compress files or folders on NTFS volumes. Instead, compress NTFS files and folders by using the Compact utility or by setting or clearing the **Compressed** attribute in My Computer or Windows NT Explorer. See Chapter 18, "Choosing a File System," in the *Windows NT Workstation Resource Guide*, for information about using My Computer or Windows NT Explorer.

Convert: Converts a Volume From FAT to NTFS

You can use the Convert utility to convert a volume from the FAT file system to the NTFS file system. This utility performs the conversion within the existing volume. You do not need to back up and restore the files when you use this utility.

You cannot convert the Windows NT boot partition while you are running Windows NT. Therefore, **convert** gives you the choice of converting the partition the next time you start Windows NT. When you convert the partition this way, Windows NT restarts twice to complete the conversion process.

To use this utility, type **convert** with the appropriate options (explained in the table following the syntax) at the command line:

convert *drive*: **/FS:NTFS [/v]**

Option	Description
drive	Logical drive that you want to convert.
/FS	You must specify that you want to convert to the NTFS file system.
/v	Run the utility in verbose mode.

DirUse: Provides Information About Usage of Disk Space

You can use the DirUse utility on the *Windows NT Server Resource Kit* CD to obtain disk space usage per folder. For more information about DirUse, type **diruse /?** at the command line, or see the Windows NT Resource Kit Tools Help for information.

This utility is useful to get the actual usage of space for compressed files and folders in NTFS volumes.

The format of the command is:

diruse [**/s** | **/v**] [**/q:***#*] [**/m** | **/k** | **/b**] [**/a**] [**/l**] [**/d**] [**/o**] [**/c**] [**/,**] [**/***] [*dirs*]

The option of interest for compressed folders and files is **/c**, which causes the display of compressed file or folder size instead of apparent size. For example, if your D drive is an NTFS volume, type **diruse /s /m /c d:** at the command prompt.

To see compression information for an individual file, you can use My Computer or Windows NT Explorer, select the file, and select **Properties** from the **File** menu.

Expand and Expndw32: Expand Compressed Files

You can use two utilities to expand one or more compressed files from your Windows NT CD, or a file that you compressed by using **compress**.

You can use Expndw32, the Windows NT-based expand utility, in one of these ways:

- If you have installed the *Windows NT Server Resource Kit*, you can double-click the Expand Files icon in the File Tools program group.
- From My Computer or Windows NT Explorer, open the Expndw32.exe file. If you have installed the *Windows NT Server Resource Kit*, this utility is in the folder into which you installed the *Windows NT Server Resource Kit*. On the *Windows NT Server Resource Kit* CD, use the version that corresponds to the hardware platform that you are using.

For help while you are using the utility, choose the Help button or press F1.

The MS-DOS-based expand utility runs from the command line. Type the **expand** command with the appropriate options, as shown in the table:

expand [**-r**] *source* [*destination*]

Option	Description
-r	Renames expanded files.
source	Specifies the source file. The "*" and "?" wildcards can be used.
destination	Specifies the destination file or path. The destination can be a folder. If source specifies multiple files and the **-r** option is not specified, then *destination* must be a folder.

Backup Utilities

AT: Enables You to Schedule Windows NT Backup

The Windows NT Backup utility, Ntbackup, does not include functionality for scheduling unattended backups. However, by using the Schedule service with the command line capabilities of the Ntbackup utility, you can set up unattended backups.

The following procedure shows how to use the Schedule service and the Ntbackup utility to schedule an unattended backup of the entire C drive of a computer with an installed tape device. With minor modifications, these steps could also allow for scheduling unattended backups from a computer on a network to a computer with an installed tape device or other configurations.

▷ To schedule unattended backup

1. Start the Schedule service on the computer with the tape device already installed. This can be done by using the Services option in Control Panel, clicking **Schedule**, and clicking **Start**. Alternatively, you can configure the Schedule service to start every time Windows NT starts by clicking **Startup** and setting the **Startup Type** to **Automatic**.
2. Using any text editor, such as Notepad, create a command file, such as, Mybackup.cmd to perform the commands to backup the files.

   This example

   ```
   ntbackup backup c: /D "My Backup Files" /B /L "c:\backup.log"
   ```

   does the following:

   - Backs up all files on the C drive.
   - Replaces any files currently on the tape.
   - Labels the backup set "My Backup Files."
   - Backs up the local registry.
   - Logs all backup information to C:\Backup.log.
3. Using the AT command, schedule the command file (Mybackup.cmd) to run when desired. The following AT command schedules Mybackup.cmd to execute at 11:00 P.M. every Monday, Wednesday, and Friday:

   ```
   AT 23:00 /every:M,W,F MYBACKUP.CMD
   ```

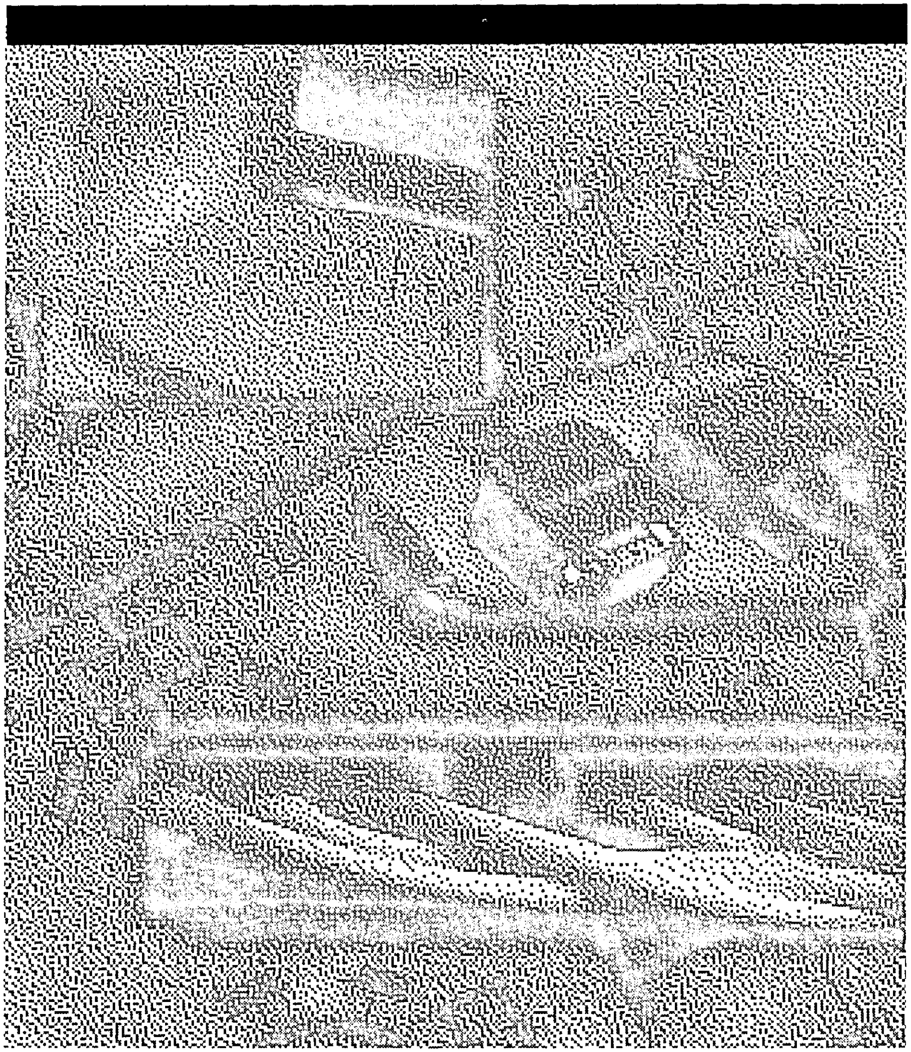
When there is no tape in the drive and you use the AT command without the interactive switch to run Windows NT Backup, the backup encounters an error or is unable to accept the command line. As a result, Windows NT Backup stops responding. You cannot run Windows NT Backup again until you restart Windows NT.

Use the /INTERACTIVE switch with the AT command to open the interactive desktop. This way, if any errors occur, you will be able to correct them and continue, or quit Windows NT Backup.

For information about AT options, type AT /? at a command prompt or search the online Command Reference for AT. For additional information on available Ntbackup options, use the Help menu when running Ntbackup.exe.

PART III

Windows NT Server Troubleshooting



CHAPTER 8

General Troubleshooting

This chapter identifies tools that are available in Windows NT to help you troubleshoot problems. It contains information about troubleshooting hardware problems, and how to use information in the Registry to determine why services aren't working correctly. It also contains an example of using information in the Registry for troubleshooting.

Careful record keeping is essential to successful troubleshooting. You should have records of your network layout, cabling, previous problems and their solutions, dates of installation of hardware and software, and so on, all readily accessible.

Many problems can be avoided with routine virus checks. Be sure to check for viruses before installing or upgrading Windows NT on a computer that is already in use.

This chapter gives information about other chapters with troubleshooting help; it explains methodology; provides an overview of Windows NT tools; describes hardware problem-solving; and explains how to identify which services or drivers are working.

Sources of Troubleshooting Information

In addition to the troubleshooting tools that are described in this chapter, there are several other sources of troubleshooting information:

- Chapter 6, "Troubleshooting Startup and Disk Problems," discusses what you can do to find the cause of problems when your computer fails to complete startup.
- The following chapters are in the *Windows NT Workstation Resource Guide*:
  - Chapter 23, "Overview of the Windows NT Registry," describes how to use information in the Registry for troubleshooting and configuration maintenance.
  - Chapter 25, "Configuration Management and the Registry," provides problem solving techniques using the Registry.
  - Chapter 39, "Windows NT Debugger," describes the different types of STOP messages. It also contains information about using the Windows NT Debugger.
- The Messages database, included in this Resource Kit, is another source of troubleshooting information. Here, thousands of messages are documented, with the probable cause and recommended solution to each of them. In particular, the Kernel STOP errors that appear when the system fails (with a blue screen) are documented in the Messages database. Chapter 38 in the *Windows NT Workstation Resource Guide*, "Windows NT Executive Messages," discusses the various types of messages generated by the Windows NT Executive, and categorizes them by their type and severity.
- Windows NT Help contains a troubleshooting topic.
- Appendix A in the *Windows NT Server Start Here* book describes how to overcome problems installing Windows NT 4.0 on x86-based computers.
- The *Windows NT Server Start Here* book describes Microsoft's AnswerPoint Information Services, which provides easy telephone access to the latest technical and support information for Microsoft products.
- The Microsoft Knowledge Base contains support information developed by Microsoft product support specialists. You can search for all Windows NT troubleshooting articles by specifying **winnt** and **tshoot** in the query. The Windows NT Knowledge Base is included on the *Windows NT Server Resource Kit* CD. It is also included in the:
  - Microsoft Developer Network (MSDN) CD.
  - TechNet CD.
  - Microsoft Internet FTP host, ftp.microsoft.com.
  - Technical Support and Services category on the Web page http://www.microsoft.com/server.

Troubleshooting Methodology

This section discusses approaches for solving a problem and presents an example of a troubleshooting scenario. There are three parts to this methodology:

- Isolating the problem.
- Identifying whether the action has ever worked properly on this computer, or works on another computer.
- Defining an action plan.

Isolating the Problem

First, try to isolate the problem. What, precisely, is not working correctly? Try to narrow down exactly what you expect to have happen versus what is happening.

For example, if your computer does not complete startup, you need to identify how far it gets, and write down any error messages. On an x86-based computer, if you get an error such as **Missing operating system** from the system BIOS when you start your computer, the problem is very different than if startup fails after the boot loader (NTLDR) starts. You know that the NTLDR has started when you see the message

```
NTDETECT V1.0 Checking Hardware . . .
```

Another way to isolate the problem is to figure out if there are related programs or functionality that works correctly on this computer. If so, what are the differences between what works and what does not work?

Identifying Whether It Works in Other Situations

Has what you are trying to do ever worked on this computer before? If so, something might have changed that affects it. Have you changed hardware or installed new software? Has somebody else been using the computer, and could that person have made changes you do not know about?

If this program or functionality has never worked on this computer, compare the setup and configuration on this computer with the same program on another computer to identify differences.

As an example, identical 624 MB IDE disks are installed on two different x86-based computers. On one computer, 609 MB are available after creating partitions. On the other computer, only 504 MB are available. If you look at the messages that the system BIOS displays when starting up the two computers, you may see that the computer with 609 MB has a newer BIOS than the other computer. You would need to upgrade the other computer's system BIOS, or obtain a third-party translation utility that enables the computer to access the entire disk.

Defining an Action Plan

Try to identify all of the variables that could affect the problem. As you troubleshoot the problem, try to change only one of these variables at a time. Keep records of what you do and the effect of each action.

It's advantageous to develop your plan on paper. Decide what steps you want to take, and what you expect to do based on the results of each step. Then do the steps in order, and follow your plan.

If you see a result for which you have no plan:

- Go back to the isolation phase.
- Identify what happens in similar situations.
- Define another plan.

Troubleshooting Scenario

Here is a scenario that shows applying this approach to an actual problem. A user was trying to upgrade a home computer to a newer version of Windows NT 4.0 (before the final product was available). The user was about half finished with copying files from the CD to the hard disk when a message appeared saying that a file could not be copied. This was how the user isolated the cause of the problem.

The user has successfully installed earlier versions of the software on this computer. Since the last upgrade of Windows NT, the CD had been used to install another program, with no problems.

Nothing has been changed on the computer since the last upgrade, except installing the other program. That program should have no relationship to the problem. Other people can install the same version of Windows NT from CD on similar computers.

The user noticed that the CD-ROM drive made noises like it was spinning faster and then slower just before the error message.

These are the steps that he used to identify and recover from the problem.

Step 1. Check the event log to see if there are any errors logged. The CD ROM device was reporting bad blocks on the CD, so Windows NT knew that there were problems.

Step 2. Inspect the CD for dust or scratches. There were no obvious problems on the CD, and the user previously had no problems using the CD.

Step 3. Copy files from the CD manually rather than running Windows NT Setup. The file that caused the error copied fine, but other files on the CD could not be copied.

Step 4. Get another CD of the same version and try to install Windows NT from it. (Perhaps there is a problem with the CD itself.) Windows NT Setup failed on the same file on both CDs, and manually copying files fails on the same files.

Step 5. Install software from other CDs that have worked on this computer before. The user noted that some work, some do not. The ones that do not work have more data on them than ones that install successfully. Therefore, something must be wrong with accessing data on the later tracks of the CD. Data is recorded on CDs starting on the innermost track. CDs vary their spin rate when reading inner versus outer tracks. Something might be wrong with the motor synchronization spin rate.

Step 6. Look inside the CD-ROM drive for signs of dust or hair that might interfere with proper operation at one end of the read head's range of motion. A hair was found stuck to the read head.

Using Troubleshooting Tools

This section provides a brief overview of the troubleshooting tools that are available on the Windows NT Server product CD and the *Windows NT Server Resource Kit* CD.

Windows NT Tools

These tools are installed when you install Windows NT Server:

Tool	Purpose	For more information:
Event Viewer	Display the system, security, and application logs.	Chapter 37, "Monitoring Events," in the *Windows NT Workstation Resource Guide.*
Performance Monitor	Measure your computer's efficiency, identify and troubleshoot possible problems, and plan for additional hardware needs.	Chapter 10, "About Performance Monitor," in the *Windows NT Workstation Resource Guide.* Chapter 8, "Monitoring Performance" in the *Windows NT Server Concepts and Planning* book.
Task Manager	Monitor active applications and processes on your computer, and start and stop them.	Chapter 11, "Performance Monitoring Tools," in the *Windows NT Workstation Resource Guide.*
Windows NT Diagnostics	Enables you to view hardware information in the Registry, such as currently loaded device drivers and IRQ values.	"Using the Windows NT Diagnostics Administrative Tool," presented later in this chapter.
Windows NT Hardware Detection Tool (NTHQ)	Identifies installed hardware and settings for diagnostic purposes.	"Using the Windows NT Hardware Detection Tool (NTHQ)," presented later in this chapter.
Network Monitor	Determines problems during session initialization, problems related to broadcast storms, and troubleshooting packets for transmission problems between computers.	Chapter 10, "Monitoring Your Network," in the *Windows NT Server Concepts and Planning* book.

Windows NT Server Resource Kit Tools

The *Windows NT Server Resource Kit* contains many tools that can be used for troubleshooting. For information about all of the tools available in the *Windows NT Server Resource Kit,* refer to the online Resource Kit Tools Help (Rktools.hlp) and double-click each of the tools groups from the Contents page.

Troubleshooting Hardware Problems

There are three Microsoft products that you can use to help troubleshoot hardware problems:

- Hardware Compatibility List (HCL)
- Windows NT Hardware Detection Tool (NTHQ)
- Windows NT Diagnostics Administrative Tool

Using the Hardware Compatibility List (HCL)

The most common cause of hardware problems is the use of hardware that is not listed on the Hardware Compatibility List (HCL). The HCL included in the *Windows NT Server Resource Kit* lists the hardware components that have been tested and have passed compatibility testing with Windows NT version 4.0. It is especially important for you to refer to the HCL if you plan to use any modems, tape backup units, and SCSI adapters.

The latest HCL is available on:

- The Web at http://www.microsoft.com/ntserver/
- Microsoft's FTP server at ftp://microsoft.com/bussys/winnt/winnt_docs/hcl

To avoid problems, make sure that you are using a device make and model that is listed on the HCL. If several models from one manufacturer are included in the HCL, only those models are supported; a slightly different model might cause problems. Where special criteria are required for a model to be supported (for example, if a particular version of driver is required), this information is described as a footnote in the HCL. As additional hardware is tested, the HCL is updated, in this way new device drivers and other system components are added to the HCL. The updated list and software are available through the electronic services listed at the end of the HCL.

Using the Windows NT Hardware Detection Tool (NTHQ)

NTHQ is an MS-DOS-based program. The next procedure describes how to run the program.

▷ **To run NTHQ**

1. When running Windows NT, insert a blank 3.5-inch floppy disk in the drive.
2. Run Makedisk.bat from the \Support\Hqtool directory on the Windows NT Server product CD.
3. Leave the floppy disk inserted, and shutdown your computer.
4. When your computer restarts, you are running NTHQ.

The file Readme.txt on the floppy disk contains details about NTHQ. You can see the same information by clicking the Help button on the NTHQ screen.

These are the three ways that NTHQ is most often used:

- Print hardware information, save it to a file, and keep the report and file with the other configuration information for your computer. This report can be used when planning to change the configuration.
- If you are having problems installing Windows NT, you can start NTHQ from the floppy disk and use it for troubleshooting, The Readme.txt file contains troubleshooting tips for installation problems.
- If you cannot start Windows NT, or have installed new hardware and cannot access it, NTHQ might help you troubleshoot the problem. Because NTHQ enables you to view the hardware that it detects, you can find out if any devices are not being detected. For example, if you have not changed your configuration since the last time you produced an NTHQ report, run a new one and compare the results. If you find a difference, you might have a hardware problem.

Using the Windows NT Diagnostics Administrative Tool

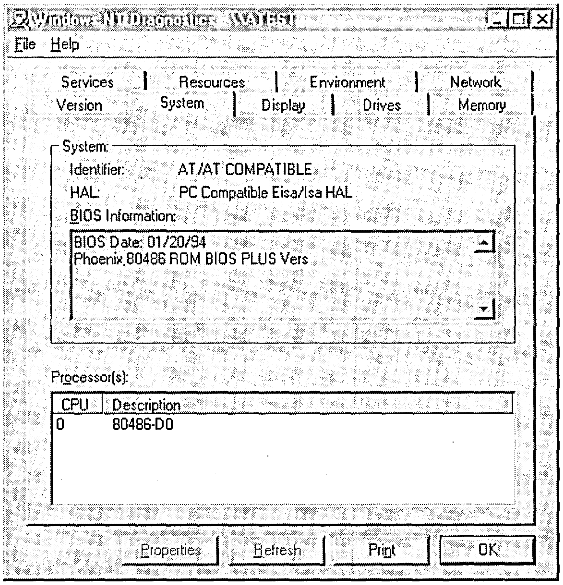
You can use this program to display Registry information in an easily-readable format. The Windows NT Diagnostics Administrative Tool enables you to:

- View information about the hardware connected to the computer.
- Identify device drivers and services that should be started when you start the computer.

▹ To run Windows NT Diagnostics

1. Click the **Start** button.
2. Click **Programs**.
3. Click **Administrative Tools (Common)**.
4. Double-click **Windows NT Diagnostics**.

The information that you can view is organized into nine tabs. The next screen shot shows the kind of information that you see when you click the **System** tab.



System:

Identifier: AT/AT COMPATIBLE

HAL: PC Compatible Eisa/Isa HAL

BIOS Information:

```
BIOS Date: 01/20/94
Phoenix 80486 ROM BIOS PLUS Vers
```

Processor(s):

CPU	Description
0	80486-D0

These are the tabs that you can select in Windows NT Diagnostics:

- **Version** shows operating system and hardware information, such as the number and type of processors.
- **System** displays more details about the computer, including the type of processor and the Hardware Abstraction Layer (HAL). The **System** tab is shown in the preceding screen shot.
- **Display** describes the video display and its driver.
- **Drives** displays information about the disks connected to the computer, the partitions and logical drives on hard disks, and network shares.
- **Memory** displays physical memory and page file information.
- **Services** provides the status of all services and devices. It provides more information than is available by using the Services and Devices options in Control Panel.
- **Resources** displays information about IRQs, I/O ports, DMA channels, and memory addresses.
- **Environment** displays system and user environment settings.
- **Network** displays user and network information.

Other Approaches to Troubleshooting Hardware Problems

If your hardware components are listed on the HCL, and you are still having problems, check that the physical connections are secure.

If you are using a SCSI device, check it termination. Even if you are sure the termination is correct, and you are having problems that could be due to incorrect termination, open the computer case and check again. You should use active rather than passive terminators whenever possible.

Note Terminators are used to provide the correct impedance at the end of a cable. If the impedance is too high or too low, internal signal reflections can take place. These echoes represent noise on the cable, and can corrupt subsequent signals, which can result in degraded performance or data loss.

Passive terminators are resistors with the appropriate resistance value for the characteristic impedance of the cable. Active terminators are slightly more sophisticated electronics that are able to better maintain the correct impedance necessary to eliminate signal reflection.

Verify that the SCSI cables are not longer than they need to be. If a two-foot cable is long enough to connect the device to the controller, do not use a three-foot cable just because you have one available. The acceptable lengths vary depending on such factors as: whether you are using basic SCSI, SCSI-2, wide SCSI, ultra-wide SCSI, differential SCSI; the quality of the termination; and the quality of the devices being used. Consult your hardware documentation for this information.

Check your hardware configuration. I/O and interrupt conflicts that went unnoticed under another operating system must be resolved when you switch to Windows NT. Likewise, you must pay much closer attention to CMOS and EISA configuration parameters when using Windows NT.

The Knowledge Base is a good source of information for hardware problems. There are several articles about memory problems, memory parity errors, SCSI problems, and other hardware information in the Knowledge Base.

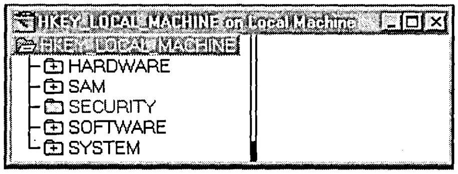
If your computer crashes randomly and inconsistently, you might have memory problems. On x86-based computers, you can use the /maxmem switch in your Boot.ini file to troubleshoot memory problems. Chapter 6, "Troubleshooting Startup and Disk Problems," contains more information about the /maxmem switch and video problems.

Troubleshooting Using HKEY_LOCAL_MACHINE

Problems can often be traced to services, device drivers, or startup control data. The Registry key HKEY_LOCAL_MACHINE contains this configuration information, so it is a good place to look for information to solve these types of problems. You have two Registry editors that you can use to look at information in the Registry:

- Regedt32.exe has the most menu items, and more choices for the menu items. You can search for keys and subkeys in the Registry.
- Regedit.exe enables you to search for strings, values, keys, and subkeys. This feature is useful if you want to find a specific value or string.

Most of the examples in this section use the Regedt32.exe. You see the following screen when you run Regedt32.exe:



The following table briefly describes the Registry keys.

Key	Description
HARDWARE	Describes the physical hardware in the computer, the way device drivers use that hardware, and mappings and related data that link Kernel-mode drivers with various user-mode code.
SAM	Contains security information for user and group accounts.
SECURITY	Contains the local security policy, such as specific user rights.
SOFTWARE	Describes the per-computer software.
SYSTEM	Controls system startup, device driver loading, Windows NT services, and operating system behavior.

The HARDWARE and SYSTEM keys are the most useful for troubleshooting.

Note Do not change information in the Registry when you are using it for troubleshooting. Instead, use the options in Control Panel, such as Services, Devices, Network, and SCSI Adapters, to change Registry information.

The Registry information and examples in this section are for a Windows NT Server computer that uses the TCP/IP network protocol. It uses a DHCP server to get IP addresses. If your computer has a different configuration, or has third-party device drivers or services installed, the Registry will contain different information.

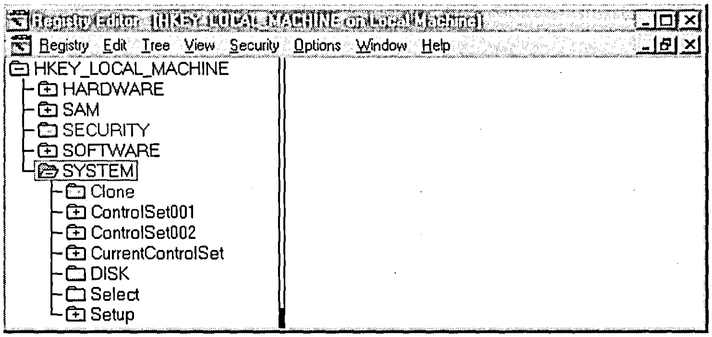
HKEY_LOCAL_MACHINE\HARDWARE

This key describes the physical hardware in the computer. Since the data in the HARDWARE key is stored in binary form, the best way to view the data are by using Windows NT Diagnostics, one of the programs in the Administrative Tools (Common) program group. See the section titled "Using the Windows NT Diagnostics Administrative Tool," presented earlier in this chapter, for more information about the program.

For more information about the HKEY_LOCAL_MACHINE\HARDWARE key, see Chapter 23 in the *Windows NT Workstation Resource Guide*, "Overview of the Windows NT Registry."

HKEY_LOCAL_MACHINE\SYSTEM

The HKEY_LOCAL_MACHINE\SYSTEM key contains information that controls system startup, device driver loading, Windows NT services, and operating system behavior. All startup-related data that must be stored (rather than computed during startup) is saved in the SYSTEM key. This screen shot shows the SYSTEM key and its subkeys.



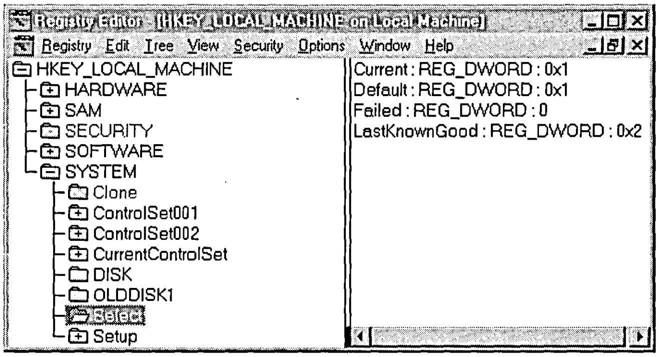
The most important troubleshooting information in the Registry key HKEY_LOCAL_MACHINE\SYSTEM are the control sets. A control set contains system configuration information, such as which device drivers and services to load and start. There are at least two control sets, and sometimes more, depending on how often you change system settings, or have problems with the settings you choose. The preceding screen shot shows the following control sets:

- Clone
- ControlSet001
- ControlSet002
- CurrentControlSet

The Registry subkey HKEY_LOCAL_MACHINE\SYSTEM\Select identifies how the control sets are used, and determines which control set is used at startup. This subkey contains the following value entries:

- Current. Identifies which control set is the CurrentControlSet. When you use Control Panel options or Registry editor to change the Registry, you are changing information in the CurrentControlSet.
- Default. Identifies which control set will be used the next time you start Windows NT, unless you select Last Known Good Configuration. Default and Current typically contain the same control set number.
- Failed. The control set that was pointed to by Default when a user last started the computer by using the LastKnownGood control set.
- LastKnownGood. The control set that is a clean copy of the last control set that actually worked. After a successful logon, the Clone control set is copied to the LastKnownGood control set.

The next screen shot shows the value entries for the Select subkey.



```
Current : REG_DWORD : 0x1
Default : REG_DWORD : 0x1
Failed : REG_DWORD : 0
LastKnownGood : REG_DWORD : 0x2
```

Note The Registry editors each display the Registry in a similar way. The window on the left contains the key and subkey names. The window on the right contains *value entries*. In the preceding screen shot, one value entry is Current : REG_DWORD : 0x1. In this example, Current is the *name*, REG_DWORD is the *data type*, and 0x1 is the *value*. These terms will be used in the rest of this section.

The values for the value entries in the Select subkey identify which control set is Current, Default, Failed, and LastKnownGood. For example, a value of 0x1 indicates that you should look at ControlSet001 to find the infromation.

In the preceding screen shot, Current and Default are both 0x1. Failed is 0, and LastKnownGood is 0x2.

Therefore, ControlSet001 is the Current and the Default control set. ControlSet001 will be the one modified if you make any changes by using options in Control Panel. ControlSet001 will be used for the Default control set the next time you start the computer.

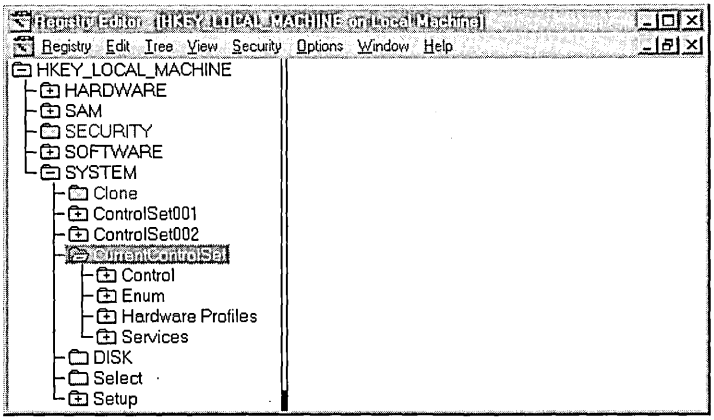
ControlSet002 is the LastKnownGood control set. If you decide to use the Last Known Good control set to start the computer, Windows NT will use ControlSet002.

For more information about the use of the control sets, see:

- Chapter 6, "Troubleshooting Startup and Disk Problems"
- Chapter 19, "What Happens When You Start Your Computer," in the Windows NT Workstation Resource Guide
- Chapter 25, "Configuration Management and the Registry," in the Windows NT Workstation Resource Guide
- The online help file Regentry.hlp on the *Windows NT Server Resource Kit* CD

Finding Service and Device Dependencies

This section describes using information in the Control and Services subkeys to troubleshoot problems with your computer. The next screen shot shows the CurrentControlSet and its subkeys.



When you install Windows NT, it creates the Control and Services subkeys for each control set in HKEY_LOCAL_MACHINE\SYSTEM. Some information, such as which services are part of which group, and the order in which to load the groups, is the same for all Windows NT computers. Other information, such as which devices and services to load when you start your computer, is based on the hardware installed on your computer and the network software that you select for installation.

Each control set has four subkeys:

- Control—Contains startup data for Windows NT, including the maximum size of the Registry.
- Enum—Contains the Plug and Play hardware tree.
- Hardware Profiles—Enables you to define different configurations for your computer and select the one you want to use at startup.
- Services—Lists all Kernel device drivers, file system drivers, and Win32 service drivers that can be loaded by the boot loader, the I/O Manager, and the Service Control Manager. It also contains subkeys describing which drivers are attached to which hardware devices, as well as the services that are installed on the system.

ServiceGroupOrder Subkey

You can see the order in which device drivers should be loaded and initialized by viewing the Registry subkey HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ServiceGroupOrder. Individual drivers that are members of a service group are loaded in this order:

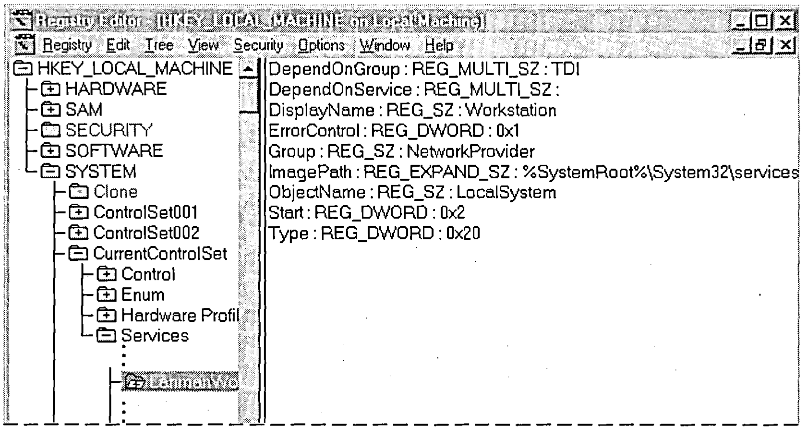
- System Bus Extender
- SCSI miniport
- Port
- Primary disk
- SCSI class
- SCSI CDROM class
- Filter
- Boot file system
- Base
- Pointer Port
- Keyboard Port
- Pointer Class
- Keyboard Class
- Video Init
- Video
- Video Save
- File system
- Event log
- Streams Drivers

- PNP_TDI
- NDIS
- TDI
- NetBIOSGroup
- SpoolerGroup
- NetDDEGroup
- Parallel arbitrator
- Extended base
- RemoteValidation
- PCI Configuration

"Service Groups," presented later in this chapter, lists drivers that are in each group.

Services Subkey

The Registry subkey HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Service name controls how services are loaded. This section describes some of the value entries for this subkey, with an explanation of their values. The next screen shot shows the subkey HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanWorkstation and its value entries.



```
DependOnGroup : REG_MULTI_SZ : TDI
DependOnService : REG_MULTI_SZ :
DisplayName : REG_SZ : Workstation
ErrorControl : REG_DWORD : 0x1
Group : REG_SZ : NetworkProvider
ImagePath : REG_EXPAND_SZ : %SystemRoot%\System32\services
ObjectName : REG_SZ : LocalSystem
Start : REG_DWORD : 0x2
Type : REG_DWORD : 0x20
```

Figure 8.1. The Registry subkey HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanWorkstation

DependOnGroup

When a subkey has a value for the DependOnGroup value entry, at least one service from the group must be loaded before this service is loaded. This table shows services that have a value for DependOnGroup. The LanmanWorkstation service, shown in Figure 8.1, has a value for the DependOnGroup value entry.

Service	Depends on
Cdfs	SCSI CDROM Class
Cdrom	SCSI miniport
Disk	SCSI miniport
LanmanServer	TDI
LanmanWorkstation	TDI
LmHosts	Network Provider
NetBIOS	TDI
Parallel	Parallel arbitrator
Scsiprnt	SCSI miniport
Scsiscan	SCSI miniport
Sfloppy	SCSI miniport

DependOnService

This value entry identifies specific services that must be loaded before this service is loaded. The "Troubleshooting Example," presented later in this chapter, shows how you can use information in the DependOnService value entry to determine which services need to be started.

This table lists the services on the example computer that have a value for DependOnServices.

Service	Depends on		
Alerter	LanmanWorkstation		
Browser	LanmanWorkstation	LanmanServer	LmHosts
ClipSrv	NetDDE		
DHCP	Afd	NetBT	TCP/IP
Messenger	LanmanWorkstation	NetBIOS	
NetBT	TCP/IP		
NetDDE	NetDDEDSDM		
NetLogon	LanmanWorkstation	LmHosts	
Parallel	Parport		
Replicator	LanmanServer	LanmanWorkstation	

By knowing the dependencies, you can troubleshoot a problem more effectively. For example, if you stop the Workstation service, the Alerter, Messenger, and Net Logon services are also stopped, because they are dependent upon the Workstation service. If an error occurs when you try to start the Workstation service, any of the files that are part of Workstation service could be missing or corrupt. This is also why, if you start one of the services that depend on Workstation service, the Service Control Manager will automatically start the Workstation service if it is not already running.

ErrorControl

This value entry controls whether an error during the startup of this driver will cause the system to switch to the LastKnownGood control set. If the value is 0 (ignore, no error is reported) or 1 (normal, error reported), startup proceeds. If the value is 2 (severe) or 3 (critical), an error is reported and LastKnownGood control set will be used.

The ErrorControl value for LanmanWorkstation is 0x1, which indicates that if there was an error starting LanmanWorkstation, an error would be logged in the event log, but Windows NT would complete startup.

ImagePath

This value entry identifies the path and file name of the driver. You can use My Computer or Windows NT Explorer to verify the existence of the named file. The ImagePath for LanmanWorkstation is %SystemRoot%\system32\services.exe.

Start

This value entry determines when services are loaded during system startup. If a service is not starting, you need to know when and how it should be starting. Then look for the services that should have been loaded prior to this service. The values are described as follows:

Value	Meaning	Description
0	Boot	Loaded by the boot loader (NTLDR or OSLOADER) during the startup sequence.
1	System	Loaded at Kernel initialization during the load sequence.
2	Auto Load	Loaded or started automatically at system startup.
3	Load On Demand	Driver is manually started by the user or another process.
4	Disabled	Driver is not to be started under any condition. If a driver is accidentally disabled, reset this value by using the Services option in Control Panel. File System drivers are the one exception to the Start value. They are loaded even if they have a start value of 4.

Type

The Type value entry helps you know where the service fits in the architecture. These are its possible values:

Value	Description
0x1	Kernel device driver.
0x2	File System driver, which is also a Kernel device driver.
0x4	Set of arguments for an adapter.
0x10	A Win32 program that can be started by the Service Controller and that obeys the service control protocol. This type of Win32 service runs in a process by itself.
0x20	A Win32 service that can share a process with other Win32 services.

Many of the services that have a Type value of 0x20 are part of the Services.exe. For example, if your network protocol is TCP/IP, and you are configured to use a DHCP server to get IP addresses, these services that have a Type value of 0x20 are in the Services.exe:

- Alerter
- Browser
- DHCP
- EventLog
- LanmanServer
- LanmanWorkstation
- LmHosts
- Messenger
- NtLmSsp
- PlugPlay

These services are part of the Netdde.exe:

- NetDDE
- NetDDEdsdm

Services Groups

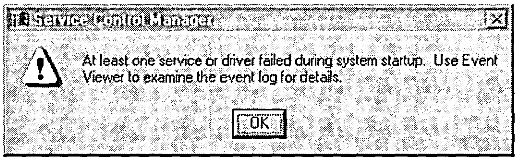
Many device drivers are arranged in groups to make startup easier. When device drivers are being loaded, Windows NT loads the groups in the order defined by ServiceGroupOrder. The next table shows which drivers are in each group.

Group name	Services		
BASE	Beep	KSecDD	Null
Boot Files System	Fastfat	Fs_Rec	
Event log	EventLog		
Extended Base	Modem Parallel	Scsiprnt	Serial
File System	Cdfs Msfs	Npfs	Ntfs
Filter	Cdaudio Changer	Diskperf Ftdisk	Simbad
Keyboard Class	Kbdclass		
Keyboard Port	i8042prt		
NDIS	EE16	NDIS	
NetBIOSGroup	NetBIOS		
NetDDEGroup	NetDDE		
Network	Mup	Rdr	Srv
NetworkProvider	LanmanWorkstation		
Parallel Arbitrator	Parport		
PCI Configuration	PCIDump		
PlugPlay	PlugPlay		
Pointer Class	Mouclass		
Pointer Port	Busmouse	Inport	Sermouse
Port	none		
PNP_TDI	NetBT	Tcpip	
Primary Disk	Abiosdsk Atdisk	Floppy	Sfloppy
RemoteValidation	NetLogon		
SCSI CDROM Class	Cdrom		

Group name	Services		
SCSI Class	Disk	Scsiscan	
SCSI Miniport	Aha154x Aha174x aic78xx Always ami0nt amsint Arrow atapi BusLogic Cpqarray dac960nt dce376nt	Delldsa DptScsi dtc329x Fd16_700 Fd7000ex Fd8xx mitsumi mkecr5xx Ncr53c9x Ncrc700 Ncrc710 ncrc810	Oliscsi Ql10wnt slcd32 Sparrow Spock T128 T13B tmv1 Ultra124 Ultra14f Ultra24f Wd33c93
SpoolerGroup	Spooler		
Streams Drivers	none		
System Bus Extender	Pcmcia		
TDI	Afd	DHCP	
Video	Ati Cirrus Dell_DGX Et400 Jazzg30 Jazzg364 Jzvxl484	mga mga_mil ncr77c22 psidisp qv s3 tga	v7vram VgaSave wd90c24a wdvga weitekp9 Xga
Video Init	VgaStart		
Video Save	VgaSave		

Troubleshooting Example

This section describes using information in the DependOnGroup and DependOnService to find the cause of the following error message that you see after you log on.



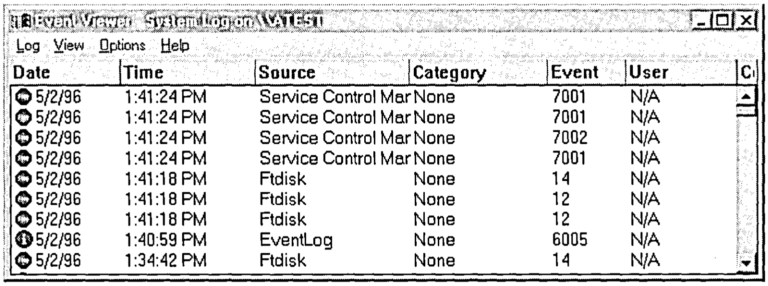
At least one service or driver failed during system startup. Use Event Viewer to examine the event log for details.

You can use the Event Viewer to see which services or drivers did not start.

▷ To run Event Viewer

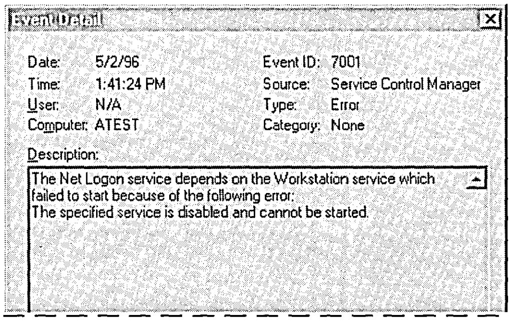
1. Click the **Start** button.
2. Click **Programs.**
3. Click **Administrative Tools (Common).**
4. Double-click **Event Viewer.**
5. If the screen is displaying a log other than System Log, on the **Log** menu, click **System.**

The event log shows the following entries:



Date	Time	Source	Category	Event	User	C
5/2/96	1:41:24 PM	Service Control Mar	None	7001	N/A	
5/2/96	1:41:24 PM	Service Control Mar	None	7001	N/A	
5/2/96	1:41:24 PM	Service Control Mar	None	7002	N/A	
5/2/96	1:41:24 PM	Service Control Mar	None	7001	N/A	
5/2/96	1:41:18 PM	Ftdisk	None	14	N/A	
5/2/96	1:41:18 PM	Ftdisk	None	12	N/A	
5/2/96	1:41:18 PM	Ftdisk	None	12	N/A	
5/2/96	1:40:59 PM	EventLog	None	6005	N/A	
5/2/96	1:34:42 PM	Ftdisk	None	14	N/A	

Sometimes, as you can see by the preceding System Log screen shot, several events are logged at approximately the same time. In this example, the newest event is entered at the top. Usually, if you look at the oldest event, you will find the reason that all of the events are logged. In the preceding example, the fourth entry from the top was the first one logged at 1:41:24. Double-clicking on it results in this event detail.



Date: 5/2/96
Time: 1:41:24 PM
User: N/A
Computer: ATEST
Event ID: 7001
Source: Service Control Manager
Type: Error
Category: None

Description:

The Net Logon service depends on the Workstation service which failed to start because of the following error:
The specified service is disabled and cannot be started.

But if you look in the Registry there is no subkey HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Workstation. You have two methods that you can use.

You can use Regedit.exe to find the name anywhere in the control set.

▷ To use Regedit.exe to find the Workstation service

1. Click the **Start** button.
2. Click **Run**, and enter Regedit.exe.
3. Double-click **HKEY_LOCAL_MACHINE**, double-click **SYSTEM**, double-click **CurrentControlSet**, and click **Services**.
4. On the **Edit** menu, click **Find**.
5. In the **Find what** box, enter **Workstation** and check the **Keys** and **Data** checkboxes. Clear **Match whole string** only.
6. Click **Find**.
7. If the match is not what you are looking for, on the **Edit** menu, click **Find Next** until you find the correct key.

If you think that the service name is part of the key name, you can use the Windows NT Registry Editor.

▷ To use Regedt32.exe to find the Workstation service

1. Click the **Start** button.
2. Click **Run**, and enter Regedt32.exe.
3. Double-click **HKEY_LOCAL_MACHINE**, double-click **SYSTEM**, double-click **CurrentControlSet**, and click **Services**.
4. On the **View** menu, click **Find key**.
5. In the **Find what** box, enter **Workstation**. Clear **Match whole word only** and **Match case**.
6. Click **Find Next**.

Both Registry editors find a match on the subkey HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanWorkstation. The DisplayName value entry contains the name that you see when you use the Services icon in Control Panel, or the Services tab in the Windows NT Diagnostics administrative tool, to view information about services.

Therefore, this subkey is the one you are searching for. Its Start value is 0x4, which means it is disabled. It should be set to 0x2, which indicates it would start automatically when you start Windows NT.

As it turns out, you specifically disabled the Workstation service by using the Services icon in Control Panel and setting the **Startup Type** to **Disabled**. Then, the computer was restarted to see what happened.

But what about the other errors that are in the event log? If you double-click each of the first three entries, you find the following descriptions:

```
The Messenger service depends on the Workstation service which
failed to start because of the following error.
The specified service is disabled and·cannot be started.

The Computer Browser service depends on the TCP/IP NetBIOS
Helper service which failed to start because of the following error.
The dependency group or service failed to start.

The TCP/IP NetBIOS Helper service depends on the NetworkProvider
group and no member of this group started.
```

Changing the LanmanWorkstation service to start automatically will solve the problem with the Messenger service failing to start.

The Computer Browser and TCP/IP NetBIOS errors are both the result of no member of the NetworkProvider group starting. How do you find what services are in the NetworkProvider group? Regedt32.exe doesn't have an option to search for data, so you can use the Regedit.exe to find the NetworkProvider group.

▷ To use Regedit.exe to find the NetworkProvider group

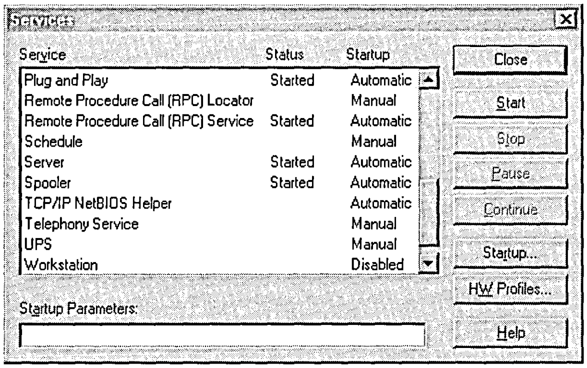
1. Click the **Start** button.
2. Click **Run**, and enter Regedit.exe.
3. Double-click **HKEY_LOCAL_MACHINE**, double-click **SYSTEM**, double-click **CurrentControlSet**, and click **Services.**
4. On the **Edit** menu, click **Find.**
5. In the **Find what** box, enter **NetworkProvider** and check the **Data** checkbox.
6. Click **Find Next.**

The only subkey that has a Group value of NetworkProvider is LanmanWorkstation. Changing LanmanWorkstation to start automatically will also solve these problems.

Identifying a Service or Driver That Doesn't Start

Some services are configured to start automatically on Windows NT. The specific services depend on your computer configuration, and which network services and protocols you are using.

You can use the Services option on Control Panel to view which services should have started automatically and see which ones did start. For example, the next screen shot was taken when the Workstation service was disabled.



Service	Status	Startup
Plug and Play	Started	Automatic
Remote Procedure Call (RPC) Locator		Manual
Remote Procedure Call (RPC) Service	Started	Automatic
Schedule		Manual
Server	Started	Automatic
Spooler	Started	Automatic
TCP/IP NetBIOS Helper		Automatic
Telephony Service		Manual
UPS		Manual
Workstation		Disabled

Startup Parameters:

You can see that TCP/IP NetBIOS Helper is configured to start automatically, but it did not start. The section "Troubleshooting Example," presented earlier in this chapter, describes why it did not start.

Sometimes, if a file that is needed to load or run Windows NT becomes corrupt or is deleted, the system displays a message about a problem with the file. You might also get information logged in the event log. You can use the message or the information in the event log to find the problem.

Not all executables or dynamic link libraries report missing or corrupt files, and the symptoms can be unpredictable with a file missing. What do you do if there is no indication of an error, but you think some component did not start correctly?

You can check to see if all the Windows NT system files exist and appear to be uncorrupted. Symptoms of file corruption include a file being an unusual size (for example, zero bytes or larger than its original size), or having a date or time that does not match the Windows NT installation date or dates on service packs that you have installed. You can compare files in your *%systemroot%*\System32 folder and subfolders with files in these folders on another computer that has the same Windows NT version and service packs installed.

If you think that you might be having a problem with a Windows NT system file, you can run Windows NT Setup and repair the problem by using the **Verify Windows NT system files** option.

If you can log on to your computer, you can use the Drivers utility on the *Windows NT Server Resource Kit* CD to display information about all the drivers that were loaded. If you have previously printed the output from the Drivers utility (by redirecting the output to a printer or a file), you can compare the previous output with one that you produce when you think you might be having problems with drivers not loading. Another method of determining if there are drivers missing from the list is to run the Drivers utility on a similar computer and compare the results.

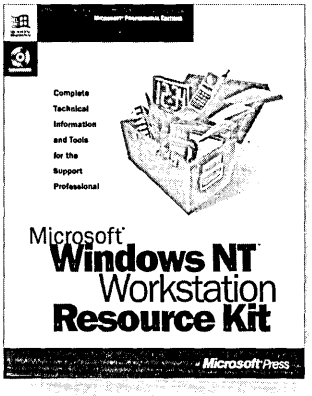
The following table is a description of the output from the Drivers utility. The most important field is ModuleName, which is the name of the component.

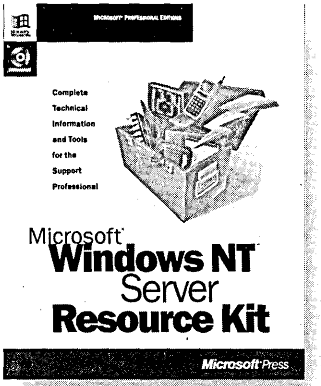
Column	Definition
ModuleName	The driver's file name.
Code	The non-paged code in the image.
Data	The initialized static data in the image.
Bss	The uninitialized static data in the image. This is data that is initialized to 0.
Paged	The size of the data that is paged.
Init	Data not needed after initialization.
LinkDate	The date that the driver was linked.

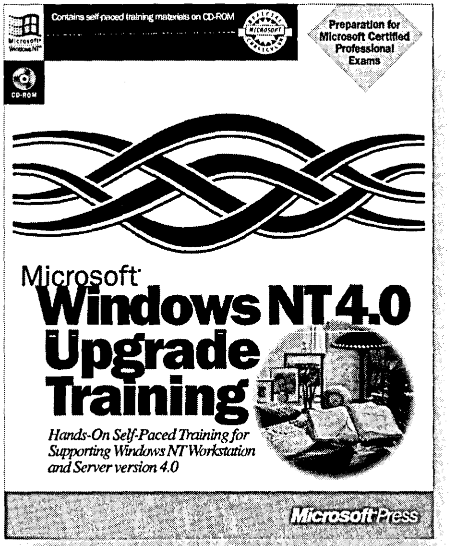
To get a hardcopy of the output from the Drivers utility enter **drivers >*filename*** at the command prompt, and then print the file. The next example shows some of the output from a Drivers report.

```
ModuleName    Code     Data  Bss   Paged   Init          LinkDate
------------------------------------------------------------------------
ntoskrnl.exe  265472  39040    0  432128  76800 Mon Apr 15 16:28:07 1996
     hal.dll   19904   2272    0    8992  10784 Wed Apr 10 10:24:22 1996
  Atdisk.sys   12352     64    0       0  10368 Wed Apr 10 10:30:24 1996
    Disk.sys    2304      0    0    7648   1504 Wed Apr 10 10:31:18 1996
  CLASS2.SYS    6112      0    0    1472   1024 Thu Apr 11 09:21:58 1996
  Ftdisk.sys   22880     32    0    1504   2048 Fri Apr 12 12:00:30 1996
Diskperf.sys    2048      0    0       0    768 Wed Apr 10 10:30:17 1996
```

Index


Complete Technical Information and Tools for the Support Professional
Microsoft Windows NT Workstation Resource Kit
Microsoft Press


Complete Technical Information and Tools for the Support Professional
Microsoft Windows NT Server Resource Kit
Microsoft Press


Contains self-paced training materials on CD-ROM
CD-ROM
Preparation for Microsoft Certified Professional Exams
Microsoft Windows NT 4.0 Upgrade Training
Hands-On Self-Paced Training for Supporting Windows NT Workstation and Server version 4.0
Microsoft Press

Microsoft® Windows NT®

Microsoft Windows NT® Server Resource Guide

For Windows NT Server Version 4.0

The MICROSOFT WINDOWS NT SERVER RESOURCE GUIDE is a supplemental source of information about Microsoft Windows NT Server version 4.0 for support professionals and network administrators.

The MICROSOFT WINDOWS NT SERVER RESOURCE GUIDE covers:

- Deploying Windows NT Server
- Printing
- System basics
- Planning a reliable configuration
- Preparing for and performing recovery
- Troubleshooting startup and disk problems
- Using disk, file system, and backup utilities
- General troubleshooting

There is no better resource for information about Microsoft Windows NT Server version 4.0 than the MICROSOFT WINDOWS NT SERVER RESOURCE GUIDE. Put it to work today.

Part No. 097-0001648

Microsoft® Press