OBJECTIVES

This chapter helps you to prepare for the Operating Systems Technologies module of the A+ Certification examination by covering the following objectives within the "Domain 2.0: Installation, Configuration, and Upgrading" section.

2.3 Identify the basic system boot sequences and boot methods, including the steps to create an emergency boot disk with utilities installed for Windows 9x/Me, Windows NT 4.0 Workstation, Windows 2000 Professional, and Windows XP.

Content may include the following:

- **Boot sequence**
- **Files required to boot**
- **Boot steps (9.x, NT-based)**
- **Alternative boot methods**
- **Using a Startup disk**
- **Safe/VGA-only mode**
- **Last Known Good Configuration**
- **Command Prompt mode**
- **Booting to a system restore point**
- **Recovery Console**
- **BOOT.INI switches**
- **Dual-booting**
- **Creating emergency disks with OS utilities**
- **Creating an emergency repair disk (ERD)**

CHAPTER 7

Starting the System

OBJECTIVES

Booting up the disk operating system is one of the most critical times in the operation of the computer. A technician must know the general sequence of events that should occur during this process. By knowing this information, the technician can observe the process and watch for telltale symptoms of operating system startup problems. Therefore, this chapter deals with booting up operating systems including Windows 9x, Windows Me, Windows NT, Windows 2000, and Windows XP.

OUTLINE

OUTLINE

STUDY STRATEGIES

To prepare for the Installation, Configuration, and Upgrading objective of the Operating Systems Technologies exam:

- ***Use all the traditional study tools we've placed in the chapter***—Pay attention to the Objectives, Challenges, and end-of-chapter questions and use them to learn the material.
- ***Use the pedagogy in this chapter to focus on the exam-specific material***—We've included lots of features geared expressly to the A+ exam. The Exam Tips scattered throughout the chapter are placed there to point to known exam-related materials. The same is true of the embedded Challenge items.
- ***Key in on Exam Tips in the chapter***—While reading through the chapter, make sure to concentrate on the following test-related items:
  - Know what types of error messages are produced when no valid MBR is found in a system.
  - Remember the files involved in the MS-DOS boot process and the order of their execution. Also, know which function keys can be used during the MS-DOS boot process to alter it.
  - Memorize the files involved in the Windows 9x startup process and the order of their execution. In particular, know which files are required to boot Windows 9x to the `Starting Windows` notice.
  - Be aware that the `BOOTLOG.TXT` file is not created during startup. It has to be initiated with the Logged mode option.
  - Know which drivers are loaded into the system when it is started in Safe Mode. Also, memorize the function keys associated with the various Safe Mode startup options.
  - Be aware that using the `WIN` switches is a viable method of starting Windows 9x systems for troubleshooting purposes.
  - Memorize the files involved in the Windows NT/2000/XP startup process and know the order of their execution.

STUDY STRATEGIES

- Be aware that the old Last Known Good Hardware Configuration settings are not replaced until a user actually logs on to the system.
- Know which types of files should be included on a typical boot disk.
- Memorize the two methods of creating emergency start disks in Windows 9x and know when these disks should be employed. Also, know which files should be present on a Windows 9x emergency start disk. In addition, you should understand why the recommended files should be added to the emergency start disk.
- Know where to make emergency start disks in Windows 9x.
- Remember that real-mode drivers must be available on the start disk to support CD-ROM drive operation in Safe Mode.
- Be aware of the function of the `Extract` command and of the file type used to store files on the Windows 9x distribution CD.
- Remember the different ways that Setup disks are created in Windows NT 4.0 and Windows 2000.
- Know what the `RDISK` command is used for in a Windows NT 4.0 system.
- Know where emergency repair disks are created in Windows NT 4.0 and in Windows 2000.
- Be aware that the Windows XP ASR function replaces the emergency repair process that was used in Windows NT and Windows 2000.

INTRODUCTION

Booting up the disk operating system is one of the most critical times in the operation of the computer. As a matter of fact, the operating system startup process is one of the major activities in which operating systems fail. Therefore, a technician must know the general sequence of events that should occur during this process. With this information, the technician can observe the process and watch for telltale symptoms of operating system startup problems.

This chapter describes basic bootup sequences associated with the MS-DOS, Windows 9x/Me, and Windows NT/2000/XP operating systems. Understanding this information becomes particularly important when the system fails to start up and you must troubleshoot it. The chapter goes on to describe alternative methods that can be used to start the system and circumvent potential problems that might be preventing the system from starting. It then concludes with explanations of how to create a Windows 9x emergency start disk with utilities installed and a Windows 2000 emergency repair disk.

After completing the chapter, you should be able to identify the major events that occur during the startup process and relate them to different groups of components in the system. You should also be able to use alternative methods to start the system to isolate potential problems that are preventing the system from starting. Finally, you should also know where and how to create a Windows 9x emergency start disk with utilities installed and a Windows 2000 emergency repair disk.

THE BOOT PROCESS

PC system boards use an IC chip to hold the system's BIOS firmware. This chip contains the programs that handle the startup of the system, the changeover to disk-based operations, video and printer output functions, and a Power On Self Test.

POST Tests and Initialization

The *Power On Self Test (POST)* is actually a series of tests that are performed each time the system is turned on. The different tests check the operation of the microprocessor, keyboard, video display,

floppy- and hard-disk drive units, as well as both the RAM and ROM memory units.

When the system is first turned on or reset, the system applies a *Reset signal* to the microprocessor and other intelligent system board components, causing them to clear most of their internal registers; however, the microprocessor sets its Instruction Pointer register to the address at the beginning of the ROM BIOS program. This is not coincidental. When the system is started up, the microprocessor must begin taking instructions from this ROM location to initialize the system for operation.

Initial POST Checks

The initial BIOS startup steps are illustrated in Figure 7.1.

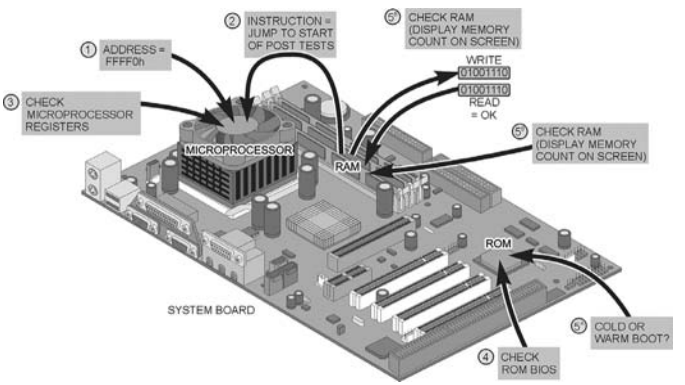


FIGURE 7.1
The startup sequence.

The first instruction that the microprocessor executes causes it to jump to the POST routines where it verifies that the BIOS program is accurate and checks the system's installed RAM (including the system's CMOS RAM). During the memory tests, the POST displays a running memory count to show that it is testing and verifying the individual memory locations. Next, the POST verifies the operation of the microprocessor's registers and performs tests on the ROM BIOS chip.

At this point, the program checks to determine whether the system is being started from an off condition or being reset from some other state. When the system is started from an off condition, a *cold boot* is being performed. Conversely, simultaneously pressing Ctrl+Alt+Del

while the system is in operation generates a reset signal in the system and causes it to perform a shortened bootup routine (some of the POST memory tests are skipped). This operation is referred to as a *warm boot*. This action permits the system to be shut down and restarted without turning it off.

System Initialization

If the system passes the initial POST, the BIOS routine initializes the system's intelligent devices. During this part of the program, startup values stored in the ROM chip are moved into the system's programmable devices (that is, Interrupt Controller, DMA Controller, chipset, and so on) to make them functional. The steps of the initialization process are depicted in Figure 7.2.

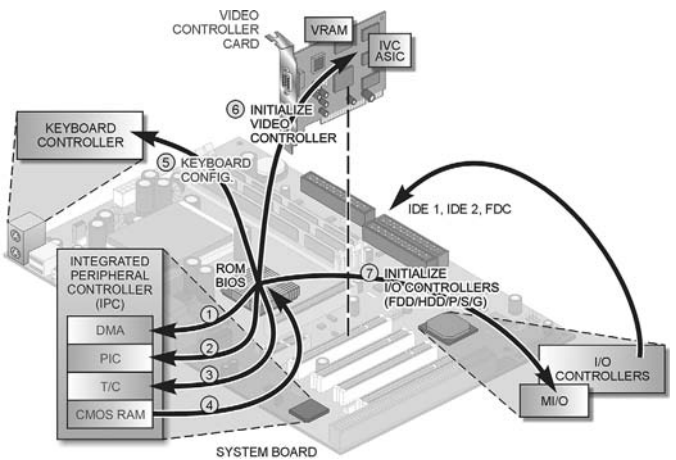


FIGURE 7.2
System initialization.

The BIOS routines read configuration information from the system's CMOS RAM to determine what type of display (VGA, SVGA, and so on) is being used with the system. After this parameter is established, the routine tests the video adapter's RAM. If it passes these tests, the system displays a cursor symbol on the monitor.

Additional POST Checks

After the system's display adapter is checked, the BIOS routines test all additional RAM on the system board and execute the system's

built-in setup program to configure its day/time setting, its hard-disk and floppy-disk drive types, and the amount of memory actually available to the system. Following the final memory test, the keyboard, the I/O devices, and adapters are tested. The steps of the additional POST process are depicted in Figure 7.3.

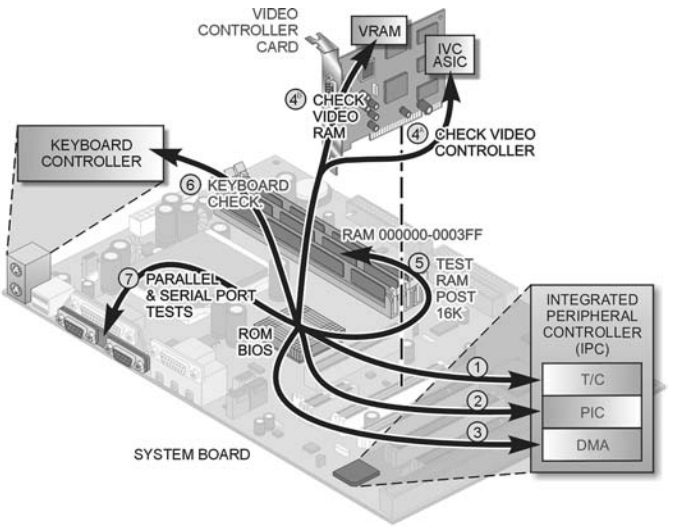


FIGURE 7.3
Completion of the POST.

After the initialization and POST tests are completed, the BIOS checks in the upper memory area (UMA) for BIOS extension programs. System designers created this memory area so that new or nonstandard BIOS routines could be added to the basic BIOS structure. These extended firmware routines match software commands from the system to the hardware they support. Therefore, the software running on the system does not need to be directly compatible with these hardware devices.

Video cards contain video BIOS code extensions, as do different types of HDD controller cards. Network adapter cards are another type of device that commonly uses BIOS extensions. These extensions enable the computer to be connected to other computers in the local area and cause the local computer to load up and operate from the operating system of a remote computer. The system can accommodate as many extensions as will mathematically fit within the allotted memory area.

At the completion of the POST and initialization operations, the BIOS routines of most systems cause the speaker circuitry to produce a single short tone. The single beep indicates that the POST portion of the bootup has been successfully completed. This is an important point in the overall troubleshooting scheme for hardware/software/configuration problems.

NOTE

You can examine the Windows 9x Plug-and-Play process by printing out its `DETLOG.TXT` file. The printout of this file, located under the `C:\` root directory, provides a step-by-step listing of the operating system's hardware-detection process. It demonstrates the order of resource allocation as well as the process of detecting and assigning resources to the system's various hardware devices.

PLUG AND PLAY

In the case of *Plug-and-Play* systems, the BIOS must also communicate with the adapter cards located in the expansion slots to determine what their characteristics are. Even in a system using a PnP-compliant operating system, such as Windows 9x/Me or Windows 2000/XP, the BIOS must also be PnP compatible before the system can recognize and manipulate system resources. When the system is turned on, the PnP devices involved in the bootup process become active in their default configuration. Other logical devices, not required for bootup, start up in an inactive mode until the operating system can configure them.

Before starting the bootup sequence, the PnP BIOS checks the devices installed in the expansion slots to see what types they are, how they are configured, and which slots they are in. It then assigns each adapter a software handle (name) and stores their names and configuration information in a RAM table. Next, the BIOS compares the adapter information to the system's basic configuration, looking for resource conflicts. If no conflicts are detected, all the devices required for bootup are activated.

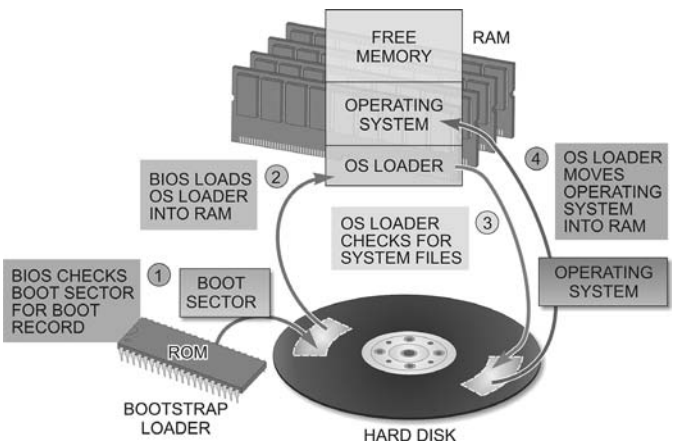
The operating system is left with the task of activating the remaining intelligent devices and resolving any resource conflicts that the BIOS detected but could not resolve. If the PnP function is not working for a particular device, or the operating system cannot resolve the remaining resource conflicts, the user needs to perform manual configurations.

FAT System Bootup

The bootup process in any system begins when the BIOS starts looking through the system for a *master boot record (MBR)*. This record can reside on drive A: or C:, or at any other location in the system in a simple single–operating system, single-disk bootup process. As you can see, this multiple-access operation uses two different bootstrap routines to locate and load two different boot records.

A simple single-operating system, single-disk bootup process is depicted in Figure 7.4. As you can see, this multiple-access operation uses two different bootstrap routines to locate and load two different boot records.

FIGURE 7.4
The bootstrap operation.



The first section on any logical disk is called the *boot sector*. This section contains information about how the disk is organized. It may also contain the small, optional master boot record that can access a larger, more powerful *Bootstrap Loader* program located in the root directory of the logical disk.

In most systems, the master boot record is located at sector 1, head-0, and track-0 of the first logical hard drive. Some texts may refer to the first sector as sector 0, keeping with the idea that the first of anything in a digital system is 0. If the disk possesses a master boot record, it can boot up the hardware system to the operating system. The disk is

then referred to as a *bootable disk*. If not, the disk is just a data disk that can be used for storing information.

Traditionally, BIOS programs searched for the master boot record in floppy-disk drive A: first. In later models, the BIOS looked first in the floppy drive or drives and then in the hard-disk drive. In newer systems, the order that the BIOS uses to search for the MBR is governed by information stored in the system's CMOS configuration RAM. The order can be set to check the floppy drive first and then the hard drive, to check the hard drive first, to check the hard drive only, or most recently, to check the CD-ROM drive.

EXAM TIP

Know what types of error messages are produced when no valid MBR is found in a system.

If the BIOS does not locate the boot record in one of the indicated drives, it will most likely display a `Non-System Disk` or `Disk Error` or `ROM BASIC Interpreter Not Found` message on the screen.

When an MBR is located, the Bootstrap Loader moves the boot record into system RAM to be executed. This record contains the Secondary Bootstrap Loader, also referred to as the *Operating System Loader*. This routine looks for an operating system boot record, typically located on the disk. When this record is found, the routine loads the bigger boot record into RAM and begins executing it. This boot record brings special operating system files into memory so that they can control the operation of the system (that is, the operating system). In the case of Microsoft FAT-based systems, the special files loaded by the OS boot record are `IO.SYS` and `MSDOS.SYS`.

In an MS-DOS system, the `IO.SYS` file executes the contents of the `MSDOS.SYS` file, looks for the MS-DOS command processor, and moves it into system RAM along with the operating system support files. The default command processor for MS-DOS is a system file called `COMMAND.COM`. This file processor provides the basic user interface called the *command line* and interprets the input entered at the command prompt.

The following list summarizes the files required to boot an MS-DOS system, along with their execution order:

EXAM TIP

Remember the files involved in the MS-DOS boot process and the order of their execution.

1. `IO.SYS`
2. `MSDOS.SYS`
3. `CONFIG.SYS`
4. `COMMAND.COM`
5. `AUTOEXEC.BAT`

Altering Bootup Steps

MS-DOS made provisions to permit the basic boot process to be modified through two special configuration files called CONFIG.SYS and AUTOEXEC.BAT. Because these files added complexity to the start-up process and often prevented systems from booting, later versions included special keystrokes that could be used to skip the execution of these files, or to move through them one step at a time for troubleshooting purposes. When these options were used, the system booted up with a complete set of default settings. No installable device drivers were installed, the current directory was set to C:\DOS, and the user often received a `Bad or missing command interpreter` message. When this message was received, the system asked the user to manually enter the path to the COMMAND.COM file.

EXAM TIP

Know which function keys can be used during the MS-DOS boot process to alter it.

The special function keys available during the MS-DOS bootup are summarized as follows:

- *F5 (also Left Shift key)*—Skips CONFIG.SYS and AUTOEXEC.BAT files.
- *F8*—Proceeds through the CONFIG.SYS and AUTOEXEC.BAT files one step at a time waiting for confirmation from the user.

WINDOWS 9X/ME STARTUP

Windows 9x/Me takes over the complete bootup function as a normal part of its operation. This seamless bootup may be convenient but can offer some interesting problems when the system will not boot: There's no stable command-line level to fall back to for troubleshooting purposes.

The Windows 9x startup sequence can be summarized as follows:

1. POST tests are run.
2. The PnP capability is configured.
3. The OS bootstrap looks for the MBR.
4. The system loads IO.SYS.
5. IO.SYS loads and executes CONFIG.SYS (if present).
6. IO.SYS loads MSDOS.SYS.

7. IO.SYS loads and executes COMMAND.COM.
8. COMMAND.COM looks for and executes AUTOEXEC.BAT (if present).
9. IO.SYS loads WIN.COM (the Starting Windows notice is displayed).
10. The KERNEL32.DLL and KERNEL386.EXE files are executed.
11. The GDI.EXE and GDI32.EXE files are executed.
12. The USER.EXE and USER32.EXE files are executed.
13. All fonts and other associated resources are loaded.
14. The WIN.INI file values are checked.
15. The Windows 9x shell and machine policies are loaded.
16. The Windows desktop components are loaded.
17. Windows 9x checks the Startup folder.

During the startup process, Windows 9x searches the Registry's HKEY_LOCAL_MACHINE key and the user's home directory for user profile information. Windows 9x creates a folder for each user who logs on to the system. This profile is held in the \Windows\Profiles subdirectory. Each profile contains a USER.DAT file (the second half of the Registry) that holds the Registry information for that user. It also contains a number of other files that customize the desktop just for that user.

As with the SYSTEM.DAT file, the USER.DAT file is backed up as USER.DA0 each time the Windows 95 system is rebooted. Under Windows 98, the USER.DAT backup is part of the RB0XX.CAB files. In Chapters 1, "Microsoft Windows Operating Systems," and 2, "Major Operating System Files," we mentioned that the USER.DAT and SYSTEM.DAT files are located in the \Windows folder.

The difference between that statement and the one above is that for single-user systems, these files are located in the \Windows folder. However, in multiple-user systems, Windows keeps profile information about all its users in the \Windows\Profiles folder. You should be aware that Windows 9x/Me does not support multiple user profiles by default. You must enable this function through the Control Panel's Users applet before the system will support multiuser settings.

EXAM TIP

Memorize the files involved in the Windows 9x startup process and the order of their execution.

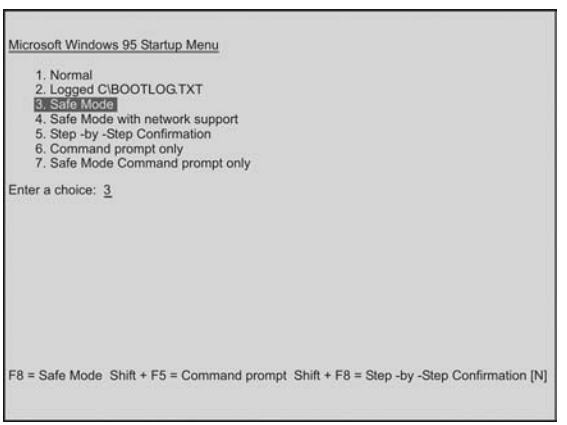
Know which files are required to boot Windows 9x to the Starting Windows notice.

Be aware of the two files that make up the different Windows 9x Registries and where they are
located.

Alternative Windows 9x Startup Modes

You can access the Windows 9x *Startup menu* (not to be confused with the desktop's Start menu), shown in Figure 7.5, on a nonstarting system by holding down the F8 function key when the `Starting Windows 9x` message is displayed onscreen. The menu offers several startup options, including Normal, Logged, Safe, Step-by-Step Confirmation, and Command Prompt modes.

FIGURE 7.5
The Startup menu.



EXAM TIP

Be aware that the `BOOTLOG.TXT` file is not created during startup. It has to be initiated with the Logged mode option.

If the *Normal* option is selected, the system tries to restart as it normally would, loading all its normal startup and Registry files. The *Logged* option also attempts to start the system in normal mode but keeps an error log file that contains the steps performed and outcome. This log is also created when the operating system determines that the previous boot attempt was unsuccessful. The text file (`BOOTLOG.TXT`) can be read with any text editor or printed out on a working system.

Safe Mode

If Windows 9x determines that a problem has occurred that prevented the system from starting, it attempts to restart the system in *Safe Mode*. This mode bypasses several startup files to provide access to the system's configuration files. You can access this startup mode by pressing the F5 function key when the `Starting Windows 9x` message is displayed onscreen.

In Safe Mode, the minimal device drivers (keyboard, mouse, and standard-mode VGA drivers) are active to start the system. The CD-ROM drive, however, is not active in Safe Mode.

In *Step-by-Step Confirmation mode*, the system displays each startup command line by line and waits for a confirmation from the keyboard before moving ahead. This way, an offending startup command can be isolated and avoided so that it can be replaced or removed. You obtain this option by pressing the F8 function key at the Startup menu.

EXAM TIP

Know which drivers are loaded into the system when it is started in Safe Mode.

Memorize the function keys associated with the various Safe Mode startup options.

NOTE

Windows 98 does not officially support Safe Mode with Network Support; however, you can access this Safe Mode version by pressing the F6 function key during Windows startup.

Command Prompt Modes

Other startup options may also be available from the menu, depending on the configuration of the system. Some options start the system and bring it to a DOS-like command prompt. Selecting the *Command Prompt Only* mode causes the system to boot up to the command line, using the startup files and the Registry.

If this option does not start the system, reboot the computer and select the Safe Mode Command Prompt Only option from the Startup menu. This option performs the same function as pressing the Shift and F5 keys simultaneously (Shift+F5) during the bootup process. The system starts in Safe Mode with minimal drivers (while not executing any of the startup files) and produces the command-line prompt.

WIN Switches

When Windows 9x/Me refuses to start up, a number of options are available for starting it from the command prompt. After gaining access to the system using a boot disk, you should attempt to start Windows by typing the `WIN` command at the command prompt. The `WIN` command can be modified by the inclusion of command switches that define the scope of the command. For example, starting Windows using a `/D` switch (that is, `WIN /D`) is often helpful in isolating possible areas of the operating system as problem sources. Note that these switches are covered in greater detail in Chapter 9, "Troubleshooting OS Startup Problems."

You can use a question mark as a switch with the `WIN` command (that is, `WIN /?`) to show a listing of all the switches associated with the

EXAM TIP

Be aware that using the `WIN` switches is a viable method of starting Windows 9x systems for troubleshooting purposes.

command. You can use these switches to start Windows with various portions of the operating system disabled. If the system runs with a particular section disabled, at least some portion of the problem can be linked to that area.

CHALLENGE #1

A client has called you from his hotel room. He is traveling to a convention, and his Windows 9x notebook computer hangs up after displaying the `Starting Windows` notice onscreen. He does not know how to begin troubleshooting the problem, and no technical support is available around the hotel. The client needs the notebook to be running tomorrow so that he can give a PowerPoint slideshow presentation at the convention. What two things can you suggest for getting past the `Starting Windows` message so the client can repair his computer?

Refer to the "Challenge Solutions" section at the end of this chapter for the resolution to the challenge.

WINDOWS NT/2000/XP STARTUP

The sequence of events in the Windows NT/2000/XP boot and startup processes is similar to that presented for FAT-based systems. The main differences are found in the terminology and filenames that Windows NT employs. Like any other PC system, the Windows NT-based PC starts up by running the POST test, performing an initialization of its intelligent system devices, and performing a system boot process. It is in the boot process that the two operating systems diverge.

When the BIOS executes the master boot record on the hard drive, the MBR examines the disk's partition table to locate the active partition. The boot process then moves to the boot sector of that partition (referred to as the *partition boot sector*) located in the first sector of the active partition. There, it finds the code to begin loading the Secondary Bootstrap Loader from the root directory of the boot drive.

In the case of an NTFS partition, the Bootstrap Loader is the NT Loader file named NTLDR. This file is the Windows NT equivalent of the FAT system's IO.SYS file and is responsible for loading the NT/2000/XP operating system into memory. Afterward, NTLDR passes control of the system over to the Windows NT operating system. The Windows 2000 and Windows XP boot processes are nearly identical to the Windows NT bootup process described in Chapter 2. The major events in the startup include

1. NTLDR looks into memory, and the `OS loader V5.0` message appears onscreen.
2. NTLDR switches the processor to 32-bit flat memory mode.
3. NTLDR starts the mini-file system (FAT or NTFS) to read disk files.
4. NTLDR reads the BOOT.INI file and displays the Advanced Boot Options menu onscreen.
5. NTLDR runs NTDETECT.COM to gather system hardware information. Ntdetect checks for key hardware items.
6. NTLDR loads the NTOSKRNL and HAL files into memory and passes the hardware information to it.
7. NTLDR reads the SYSTEM Registry key, places it in memory, and implements the hardware profile (configuration and control set) from the proper Registry.
8. NTLDR loads startup device drivers into memory.
9. NTLDR passes control to the NTOSKRNL file.
10. NTOSKRNL creates the Registry's Hardware key from the information gathered earlier by Ntdetect.
11. NTOSKRNL executes additional device drivers.
12. NTOSKRNL starts the SMSS.EXE session file.
13. The Win32 subsection runs the WINLOGON.EXE and LSASS.EXE programs, the Ctrl+Alt+Del window is presented on the display, and the logon screen is displayed.
14. The SCREG.EXE service controller program starts and loads all remaining services specified in the Registry, including the Windows 2000 shell and desktop.

As with Windows 9x, the Windows NT/2000/XP shell program is the Windows desktop. When the shell and desktop components are loaded, the system displays a prompt on the screen for the user to log in, as depicted in Figure 7.6.

FIGURE 7.6
The Windows NT Logon Information dialog box.

EXAM TIP

Memorize the files involved in the Windows NT/2000/XP startup process and know the order of their execution.

Be aware that the old Last Known Good Hardware Configuration settings are not replaced until a user actually logs on to the system.

The Windows NT/2000/XP logon enables the operating system to be configured differently for individual users. Normal logon involves entering a username and password. If no logon information is entered, default values are loaded into the system.

Alternative Windows NT 4.0 Startup Modes

Unlike the Windows 9x/Me products, Windows NT 4.0 provides very few options when it starts up. The user is normally offered two options. The NTLDR file causes the system to display a selection menu of which operating system to boot from, along with an option to start the system in VGA mode. The menu listing is based on what NTLDR finds in the BOOT.INI file. If the VGA option is selected, the system starts up as normal, with the exception that it loads only the standard VGA driver to drive the display.

The second option presented is the *Last Known Good Hardware Configuration* mode option. Selecting this option causes the system to start up using the configuration information that it recorded the last time a user successfully logged on to the system. The option appears on the screen for a few seconds after the operating system selection is made. You must press the spacebar while the option is displayed on

the screen to select this startup mode. If no selection is made, the system continues with a normal startup as previously outlined, using the existing hardware configuration information.

Alternative Windows 2000/XP Startup Modes

The Windows 2000 and Windows XP operating systems incorporate a number of Windows 9x–like startup options that can be engaged to get the system up and running in a given state to provide a starting point for troubleshooting operations. The Windows 2000 *Advanced Options Menu*, shown in Figure 7.7, contains several options that can be of assistance when you are troubleshooting startup failures. To display this menu, press F8 at the beginning of the Windows 2000 startup process.

The Windows 2000 Advanced Options Menu basically provides the

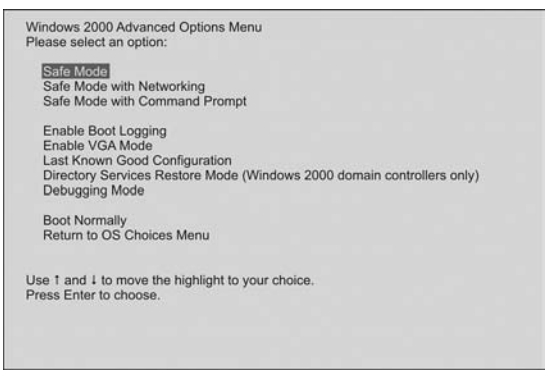


FIGURE 7.7
The Advanced Options Menu.

same Safe Mode options as the Windows 9x operating systems (that is, Boot Normally, Safe Mode, Safe Mode with Networking, and Safe Mode with Command Prompt). The Windows 2000 Advanced Options Menu also provides a number of Windows NT–like options:

- *Enable Boot Logging*—This option creates a log file called `NTBTLOG.TXT` in the root folder. This log is similar to the `BOOTLOG.TXT` file described earlier in that it contains a listing of all the drivers and services that the system attempts to load dur-

ing startup and can be useful when you are trying to determine what service or driver is causing the system to fail.

- *Enable VGA Mode*—When selected, this option boots the system normally but uses only the standard VGA driver. If you have configured the display incorrectly and are unable to see the desktop, booting into VGA Mode enables you to reconfigure those settings.
- *Last Known Good Configuration*—This option starts Windows 2000 or Windows XP using the settings that existed the last time a successful user logon occurred. All system setting changes made since the last successful startup are lost. This option is particularly useful if you have added or reconfigured a device driver that is causing the system to fail.
- *Debugging Mode*—This option starts Windows 2000 or Windows XP in a kernel debug mode that enables special debugger utilities to access the kernel for troubleshooting and analysis.

Windows 2000/XP Recovery Console

The Recovery Console available in Windows 2000 and Windows XP is a command-line interface that provides you with access to the hard disks and many command-line utilities when the operating system does not boot (that is, after the Last Known Good Configuration and Safe Mode options have been tried). The Recovery Console can access all volumes on the drive, regardless of their file system type. However, if you have not added the Recovery Console option prior to a failure, you cannot employ it and need to use the Windows 2000 Setup disks or the Windows 2000/XP distribution CD instead.

You can use the Recovery Console to copy files from a floppy disk, CD, or another hard disk to the hard disk used for booting; single-step through the startup procedure; add, remove, or format volumes on the hard disk; repair the MBR or boot sector of a hard disk or volume; and restore the Registry.

The Recovery Console can be permanently installed on a system and be made accessible from the Advanced Options Menu. You can start

the Recovery Console at any time by booting from the Windows 2000 Setup disks or the Windows 2000/XP distribution CDs, choosing to repair an installation, and selecting Recovery Console from the repair options.

To install the Recovery Console on a computer, follow these steps:

1. Put the Windows 2000 or Windows XP distribution CD in the CD-ROM drive, or connect to an installation share on the network (for example, the path on a Windows XP CD for installing the recovery console is `D:\I386\WINNT32`, where `D:\` is the CD-ROM drive).
2. Run the `winnt32 /cmdcons` command. Windows 2000/XP Setup starts up, as illustrated in Figure 7.8, and installs the Recovery Console.
3. The Recovery Console is automatically added to the Advanced Options Menu.

FIGURE 7.8
Installing the Recovery Console using `winnt32 /cmdcons`.

You can run the Recovery Console from the distribution CD for both Windows 2000 and Windows XP. To do so, start the system with the distribution CD in the drive and choose the option to Repair (Press the R key) the installation. Enter the administrator's password to access the Recovery Console. The password protection for the Recovery Console permits only two incorrect attempts by default. On the third incorrect attempt, the system stops accepting further entries for a predetermined amount of time (referred to as *lockout time*).

When you start the Recovery Console, you are prompted to choose the folder that contains the Windows 2000 or Windows XP installation that you are trying to repair and then to log on as administrator. The commands that can be used with the Recovery Console include most of the MS-DOS–based commands covered in Chapter 3, "Using Command-Line Functions." After you log on to the system, you can type `HELP` at the command line to obtain a list of available commands.

Altering BOOT.INI

The BOOT.INI file in Windows NT/2000/XP systems is a special, hidden read-only boot text file that is used to generate the Advanced Boot Options menu during the NT/2000/XP startup process. The system reads this file during the bootup process and places the Advanced Boot Options menu on the screen to permit the user to select different bootup options. If no selection is made within a specified time frame, the bootup process continues in default mode.

The settings in the BOOT.INI file are used to select an operating system to boot to. It supports the starting of different Windows NT versions as well as providing for starting one non-NT operating system. The BOOT.INI file contains two sections of text that can be read and modified. This enables technicians and administrators to configure the system so that the Boot Options menu will display the different operating system options available.

The default values of the BOOT.INI file are generated automatically by the system when the Windows NT/2000/XP operating system is installed. You can access this file using a text editor. Before doing this, you must change its attributes so that it becomes visible and so that you can open it. You can do this by using My Computer, Windows Explorer, or the Attrib command.

There are many switches available for use with the Windows NT entries in the menu to change the bootup process. Some of the switches that are particularly interesting for troubleshooting purposes are as follows:

- /BASEVIDEO forces the computer to start up using the Standard VGA driver.
- /DEBUG starts the system in Debug mode as described in the "Alternative Windows 2000/XP Startup Modes" section earlier in this chapter.
- MAXMEM:n can be used to specify the maximum amount of RAM that the system will start up with.
- /SOS displays the different drive names as they are loaded (similar to the step-by-step startup option in Safe Mode).

One of the major Windows Me/XP Recovery Console commands is Bootcfg. This command can be used to change the configuration of

the BOOT.INI file or to recover from bootup problems. The Bootcfg file is available for use only in the Recovery Console.

DUAL-BOOTING

As mentioned in Chapter 6, "Installing and Upgrading Operating Systems," in a dual-boot system a startup menu is established that can be used to boot the system into different operating systems located on the disk. Depending on which operating system option the user selects from the menu, the system retrieves the correct set of files and then uses them to boot the system.

To accomplish this task, Windows 9x swaps versions of the boot files back and forth between their standard names and a designated set of backup names. In Windows NT/2000/XP systems, you should place the operating systems in different partitions and access them through the Advanced Options Menu at startup.

If the system has been configured for dual-booting, the option to do so appears in the *Boot Options menu* (or the Advanced Options Menu in Windows 2000/XP systems), which appears whenever you boot up. This is the same startup menu that you can access by pressing the F8 function key during the boot process.

When you are installing the operating systems on a computer that will be used in a dual-boot configuration, you should install the oldest operating system version first and then work your way to the newest operating system in order (that is, Windows XP should always be last).

WINDOWS XP SYSTEM RESTORE

The Windows XP System Restore utility enables administrators to roll back the Windows XP Professional operating system to a previous operational state and configuration—without affecting any user's personal data. This feature extends the Last Known Good Configuration mode by allowing the system to be rolled back to predetermined restore points.

Restore points are records of information that are created at specific intervals and when certain events occur. Some restore points are created automatically on a 24-hour/daily basis. Others are created when significant events occur, such as when you upgrade the system hardware or software, when you perform a recovery operation, or when a new driver is loaded. However, restore points can also be created manually as a method of preserving the current state of the operating system prior to performing management activities. Such activities include

- When you are updating a driver and it appears to cause problems with the system that rolling back the driver does not resolve.
- When you are installing a new software program and it creates problems with the system that uninstalling the software does not resolve.
- Any time you need to get back to a point where you know the system was functioning correctly.

You should actually create a Restore point any time that you are making changes to the system that might make it unstable or that might disable it.

To access the System Restore Wizard through the Start menu's Help and Support option, select the Performance and Maintenance option from its menu, click the Using System Restore to Undo Changes entry, and click the Run the System Restore Wizard option.

Using the Windows XP System Restore function is described in greater detail in Chapter 9.

SELF-BOOTING DISKS

Boot disks are referred to by different names depending on which operating system you are discussing. In MS-DOS the boot disk was simply referred to as a *clean boot disk*. Under Windows 9x it is called an *emergency start disk*. Windows NT and Windows 2000 refer to these as *Setup disks*. In all the operating system versions, this tool provides a well-defined point to begin troubleshooting operating system problems.

When making a simple MS-DOS boot disk, you must install the files required to boot to an MS-DOS command prompt: IO.SYS, MSDOS.SYS, and COMMAND.COM. You can do so by placing a blank floppy in the drive and typing FORMAT A:/S as described in Chapter 3 or by typing SYS A: at the command prompt. The CONFIG.SYS and AUTOEXEC.BAT files are helpful in configuring a bootup but are not required to boot to a command prompt. However, to make the boot disk truly useful, you should include at least a few utilities that can be used to provide initial troubleshooting functions after you have gained access to the system. These functions include partitioning, formatting, file editing, and diagnostic utilities.

EXAM TIP

Know which types of files should be included on a typical boot disk.

Be aware of the two methods of creating emergency start disks in Windows 9x and know when these disks should be employed.

Creating a Windows 95 Emergency Start Disk

In the event that the Windows OS becomes nonfunctional, you need to use the emergency start disk to gain access to the system so that you can restore it to proper operation. The Windows 9x emergency start disk boots up the system only to the command prompt. From this point, you need to be familiar with command-line operations so that you can employ tools and utilities that will get the system up and running again.

During the Windows setup operation, the software provides an option for creating an emergency start disk. This option should be used for every Windows 9x installation. Setup copies the operating system files to the disk along with utilities for troubleshooting startup problems. The disk can then be used to boot up the system in Safe Mode and display a command-line prompt. The emergency start disk can also be used to replace lost or damaged system files on the hard disk.

An emergency start disk can also be created through the Control Panel's Add/Remove Programs icon. This option is normally used to create a new Startup disk after new hardware has been installed or when configuration information has been changed.

In addition to creating a startup floppy disk, Windows 95 transfers a number of diagnostic files to the disk, including the following:

- IO.SYS
- MSDOS.SYS

EXAM TIP

Know which files should be present on a Windows 9x emergency start disk.

Understand why the recommended files should be added to the emergency start disk.

- COMMAND.COM
- SCANDISK.EXE
- SYS.COM
- FDISK.EXE
- FORMAT.COM
- SCANDISK.INI
- EDIT.COM
- REGEDIT.EXE
- ATTRIB.EXE

Because the Windows 95 system settings are basically contained in the two Registry files SYSTEM.DAT and USER.DAT, it is not uncommon to back them up on the emergency start disk. This operation is performed with the RegEdit utility's Export function. The Export function can be used to save a selected branch or the entire Registry as a REG text file.

In addition to the .DAT Registry files, you may want to include copies of any CONFIG.SYS, AUTOEXEC.BAT, WIN.INI, SYSTEM.INI, and CD-ROM driver files on the emergency start disk. These files can be quite useful for maintaining compatibility with installed hardware and software applications. The CD-ROM driver (that is, MSCDEX.EXE) should be included to provide CD-ROM support for access to the utilities on the Windows distribution CD.

The Registry backup file can be used to restore the Registry to the system after a crash. Once again, this procedure involves using the RegEdit Import function to restore the Registry for use. The RegEdit Import function can be performed using the Windows-based version, or it can be conducted from the command line using the real-mode version located on the emergency start disk.

The REGEDIT /C command should not be used except for cases in which the Registry is heavily corrupted. It must have a complete image of the Registry to be used in this manner. Also, realize that Windows 9x backs up the Registry files each time it is started. There should be several iterations of the Registry files that could be renamed and copied over the existing Registry files to repair them.

The Windows 98 Emergency Start Disk

Like the Windows 95 emergency start disk, the Windows 98 version is basically a boot disk with key Windows 98 utilities included to assist in restarting the system when Windows 98 doesn't boot. This disk can be created during the installation process or by accessing the Startup Disk tab in the Control Panel's Add/Remove Programs window.

In addition to the system files required to start the system in a minimal real-mode condition, the Windows 98 emergency start disk provides a number of diagnostic programs and a trio of real-mode CD-ROM drivers (`MSCDEX.EXE` for IDE drives and `BTDCDROM.SYS` and `ASPICD.SYS` for SCSI drives) to enable the CD-ROM drive to operate in Safe Mode.

EXAM TIP

Know where to make emergency start disks in Windows 9x.

Know that real-mode drivers must be available to support CD-ROM drive operation in Safe Mode.

Be aware of the function of the Extract command and of the file type used to store files on the Windows 9x distribution CD.

CHALLENGE #2

You have started a customer's computer in Safe Mode because it has repeatedly failed to start Windows 95. After reaching Safe Mode, you put the Windows CD in the drive but cannot access it. What do you need to do to access the CD-ROM drive so that you can run repair utilities from it?

Refer to the "Challenge Solutions" section at the end of this chapter for the resolution to the challenge.

The Windows 98 emergency start disk includes real-mode SCSI CD-ROM support, provides a RAMDrive, and features a new *Extract command* (`EXT.EXE`) to work with `EXTRACT.EXE`. The Extract command is used to pull necessary files from the cabinet (`.CAB`) files on the Windows 98 distribution CD.

CHALLENGE #3

A customer has accidentally erased the `SCANDISK.EXE` file from her Windows 98 notebook computer. What is the least intrusive method that you can use to recover the file and get the ScanDisk function back in operation?

Refer to the "Challenge Solutions" section at the end of this chapter for the resolution to the challenge.

Windows NT/2000 Emergency Disks

In the Windows NT/2000 arena, technicians should have on hand two different types of troubleshooting-related disks:

- Setup disks
- Emergency repair disk (ERD)

Setup disks are the equivalent of the Windows 9x Startup disk. Windows NT 4.0 generates a three-disk set, and Windows 2000 creates a four-disk set. Unlike the Windows 9x Startup disk, the Setup disks do not bring the system to a command prompt. Instead, they initiate the Windows Setup process.

Both Windows NT 4.0 and Windows 2000 provide for an *emergency repair disk (ERD)* to be produced. The ERD is different from the Setup disks in that it is intended for use with an operational system when it crashes. It is not a bootable disk and must be used with the Setup disks or the Windows distribution CD.

Whereas the Setup disks are uniform for a given version of Windows NT, the ERD is specific to the machine from which it is created. It contains a copy of the *Security Accounts Manager (SAM)* in Windows NT and the Registry in Windows 2000. When dealing with the NT ERD, you need to manually copy the Registry files to the disk.

Windows NT/2000 Setup Disks

The *Windows NT Setup disks* basically perform three functions. They load a miniature file system into the system, initialize its drives, and start the installation process. All Windows NT Setup disks are the same for all machines running that version of the operating system. To create Setup disks under Windows NT 4.0, you must install the Windows NT distribution CD in the system and type `WINNT /ox` at the command prompt.

EXAM TIP

Be aware of the different ways that Setup disks are created in Windows NT 4.0 and Windows 2000.

Under Windows 2000, you must place the distribution CD in the drive and launch the *MakeBootDisk* utility to create the four disk images for its Windows 2000 Setup disks. You can also create Setup disks from the command prompt using the `MAKEBT32.EXE` file for Windows 2000. You also can make these disks by first selecting Start, Run, Browse, CD-ROM. Then, from the CD, select the `BOOTDISK` option, followed by the `MAKEBT32.EXE` command.

Windows NT 4.0 ERD

During the installation process, Windows NT Setup asks whether you want to create an emergency repair disk. You can also create an ERD later using the *Repair Disk program* (`RDISK.EXE`). To do so, select the Run option from the Start menu, enter the `CMD` command in the Run dialog box, and then type `RDISK` at the command prompt.

When Windows NT is installed, the Setup routine stores Registry information in the `\system32\config` folder and creates a `\repair` folder to hold key files.

EXAM TIP

Know what the `RDISK` command is used for in a Windows NT 4.0 system.

Know where ERDs are created in Windows NT 4.0 and in Windows 2000.

Windows 2000 ERD

The Windows 2000 Setup routine prompts you to create an ERD during the installation process. The ERD can also be created using the Windows 2000 Backup utility, which you can access by selecting Programs, Accessories, System Tools. The Windows 2000 ERD contains configuration information specific to the computer that will be required during the emergency repair process.

CHALLENGE #4

One of your associates has called you because he is a Windows 2000 technician, but he has been assigned to repair a Windows NT 4.0 workstation. He needs an emergency repair disk for the machine and cannot find it. Also, he is not familiar with Windows NT 4.0 and does not know where to create this disk. What can you tell him to help him out?

Refer to the "Challenge Solutions" section at the end of this chapter for the resolution to the challenge.

WINDOWS XP AUTOMATED SYSTEM RECOVERY

In the Windows XP operating system, the emergency repair disk has been replaced with an emergency startup tool called the *Automated*

EXAM TIP

Be aware that the ASR function replaces the emergency repair process that was used in Windows NT and Windows 2000.

System Recovery (ASR). The ASR tool can be used to back up and restore the system state information, along with all the files stored on the system volume. As with the Windows NT/2000 emergency repair disk, the ASR feature is considered to be the last resort to use when you have been unable to recover the system using other methods, including Safe Mode, Last Known Good Configuration mode, and the Recovery Console.

The ASR utility is a function of the `NTBackup.exe` backup utility. As with other backup options, ASR is a two-part system: backup and restore operations. ASR backups should be performed periodically to keep them up to date. On the other hand, an ASR restore operation is normally performed only in the case of a system failure. The contents of an ASR backup operation include

NOTE

Be aware that the ASR function can be used to back up and restore only *system* information. Therefore, you must make sure that you perform regular applications and data backups as well.

- The system state data
- The system services
- The system components for all disks

In addition, an ASR floppy disk is created during the backup operation; it contains additional information required for the ASR restore process. This disk contains two files:

- `Asr.sif`—Contains hard-disk, partition, and volume configuration information along with general system information.
- `Asrpnp.sif`—Contains PnP device configuration data.

Additional information about conducting ASR backups and restores is presented in Chapter 9.

CHAPTER SUMMARY

This chapter presented the steps of the basic bootup sequences for FAT-based, Windows 9x/Me, and Windows NT/2000/XP systems in detail. It also described alternative startup methods that can be used to isolate potential sources of startup problems in the different operating systems.

The discussions also included descriptions of different startup tools and disks that can be used to start up and investigate a defective system. These discussions included the Windows 9x emergency start disks and Windows 2000 emergency repair disks, as well as the Windows Me/XP System Restore utilities and the Automated System Recovery tool.

At this point, review the objectives listed at the beginning of the chapter to be certain that you understand each point and can perform each task listed there. Afterward, answer the review questions that follow to verify your knowledge of the information.

KEY TERMS

- Boot disks
- `BOOT.INI`
- `BOOTLOG.TXT`
- Bootstrap Loader
- Cabinet (`.CAB`) files
- Cold boot
- Command Prompt Only mode
- Emergency repair disk (ERD)
- Emergency start disk
- Extract command (`EXT.EXE`)
- `EXTRACT.EXE`
- F5 function key
- F8 function key
- `HAL.DLL`
- MakeBootDisk utility
- `MAKEBT32.EXE`
- Master boot record (MBR)
- `NTDETECT.COM`
- `NTLDR`
- `NTOSKRNL.EXE`
- Plug-and-Play
- Power On Self Test (POST)
- `RDISK`
- `REGEDIT /C`

CHAPTER SUMMARY

- Repair Disk program (`RDISK.EXE`)
- Reset signal
- Safe Mode
- Setup disks
- Shift+F5
- `SMSS.EXE`
- Startup menu
- Step-by-Step Confirmation mode
- Warm boot
- `WIN` command
- Windows NT Setup disks
- `WINNT /ox`

APPLY YOUR KNOWLEDGE

Review Questions

1. What type of error message is produced when no valid MBR is found in a system?

   A. `Non-System Disk`

   B. `Invalid Media Type`

   C. `No Partition Found`

   D. `Disk Not Found`

2. What is the order of execution for the files involved the MS-DOS boot process?

   A. `IO.SYS`, `MSDOS.SYS`, `COMMAND.COM`, `CONFIG.SYS`, `AUTOEXEC.BAT`

   B. `IO.SYS`, `MSDOS.SYS`, `CONFIG.SYS`, `AUTOEXEC.BAT`, `COMMAND.COM`

   C. `IO.SYS`, `MSDOS.SYS`, `CONFIG.SYS`, `COMMAND.COM`, `AUTOEXEC.BAT`

   D. `MSDOS.SYS`, `IO.SYS`, `CONFIG.SYS`, `COMMAND.COM`, `AUTOEXEC.BAT`

3. Which function key can be used to alter the MS-DOS boot process?

   A. F1

   B. F4

   C. F5

   D. F11

4. Which files are involved in the Windows 9x start-up process, and what is the order of their execution?

   A. `IO.SYS`, `CONFIG.SYS`, `MSDOS.SYS`, `COMMAND.COM`, `AUTOEXEC.BAT`, `WIN.COM`, `KERNEL32.DLL`, `KERNEL386.EXE`, `GDI.EXE`, `GDI32.EXE`, `USER.EXE`, `USER32.EXE`, fonts, `WIN.INI`

   B. `IO.SYS`, `MSDOS.SYS`, `CONFIG.SYS`, `COMMAND.COM`, `AUTOEXEC.BAT`, `WIN.COM`, `KERNEL32.DLL`, `GDI32.EXE`, `USER.EXE`, `USER32.EXE`, fonts, `WIN.INI`

   C. `MSDOS.SYS`, `IO.SYS`, `CONFIG.SYS`, `COMMAND.COM`, `AUTOEXEC.BAT`, `WIN.COM`, `KERNEL32.DLL`, `GDI32.EXE`, `USER.EXE`, `USER32.EXE`, fonts, `WIN.INI`

   D. `IO.SYS`, `MSDOS.SYS`, `CONFIG.SYS`, `COMMAND.COM`, `AUTOEXEC.BAT`, `WIN.COM`, `KERNEL32.DLL`, `KERNEL386.EXE`, `GDI.EXE`, `GDI32.EXE`, fonts, `WIN.INI`

5. When in Safe Mode, the `BOOTLOG.TXT` file has to be initiated with the _____ option.

   A. Network Support

   B. Command prompt

   C. Logged mode

   D. Step-by-Step Confirmation

6. Which items are loaded into the system when it is started in Safe Mode?

   A. `BOOT.INI`, `CONFIG.SYS`

   B. Ntdetect, `BOOT.INI`, `NTLDR`

   C. mouse, keyboard, VGA

   D. network adapter, mouse, keyboard

7. Which of these function key commands is associated with booting to the Startup menu?

   A. F5

   B. Shift+F5

   C. F6

   D. F8

APPLY YOUR KNOWLEDGE

8. Which of these function key commands is associated with booting to standard Safe Mode?

   A. F5

   B. Shift+F5

   C. F6

   D. F8

9. What command can you use from the command line to show a listing of all the Windows 9x boot-up switches?

   A. `WIN ?`

   B. `WIN /?`

   C. `WIN /HELP`

   D. `WIN /H`

10. What are the names of the files involved in the Windows NT boot process, and in what order are they executed?

    A. `BOOT.INI`, `NTLDR`, `NTOSKRNL.EXE`, `NTDETECT.COM`

    B. `NTLDR`, `BOOT.INI`, `NTOSKRNL.EXE`, `NTDETECT.COM`

    C. `BOOT.INI`, `NTLDR`, `NTDETECT.COM`, `NTOSKRNL.EXE`

    D. `NTLDR`, `BOOT.INI`, `NTDETECT.COM`, `NTOSKRNL.EXE`, `HAL.DLL`

11. The old Last Known Good Hardware Configuration settings are not replaced until _____.

    A. a user logs off the system

    B. a user shuts down the operating system

    C. a user boots the operating system

    D. a user logs on to the system

12. Which files must be present on an MS-DOS boot disk? (Select all that apply.)

    A. `IO.SYS`

    B. `MSDOS.SYS`

    C. `COMMAND.COM`

    D. `CONFIG.SYS`

13. What are the two methods for creating emergency start disks in Windows 9x? (Select two answers.)

    A. Type `Format a: /c`.

    B. In the Add/Remove Programs Properties window, select the Startup Disk tab.

    C. Select the option during installation.

    D. Right-click on the desktop and select Create System Disk.

14. Which of these files is included on the Windows 9x Startup disk?

    A. `WIN.COM`

    B. `FDISK.EXE`

    C. `PAGEFILE.SYS`

    D. `NTLDR`

15. Which file is not required to boot the Windows NT 4.0 operating system?

    A. `BOOT.INI`

    B. `NTLDR`

    C. `NTBOOTDD.SYS`

    D. `HAL.DLL`

APPLY YOUR KNOWLEDGE

16. Which file must be present on the Windows 9x Startup disk to provide CD-ROM support?

    A. MSCDDVR.EXE

    B. CDDRVR.SYS

    C. MSCDEX.EXE

    D. CDFS.SYS

17. What is the file extension used to store files on the Windows 9x distribution CD?

    A. .ZIP

    B. .EXE

    C. .CAB

    D. .TAR

18. What is the utility used to retrieve a file from a cabinet (.CAB) archive file located on the Windows 9x distribution CD?

    A. EXPAND

    B. RESTORE

    C. COPY

    D. EXT

19. Which command is used to create Setup disks in Windows NT 4.0?

    A. MAKEBT

    B. WINNTCD /O

    C. WINNT /OX

    D. MAKEBT32

20. What are two methods of creating an ERD in Windows 2000? (Select two answers.)

    A. You can create it as a part of the Windows setup process.

    B. Use the Add/Remove Programs utility.

    C. Use the Backup utility.

    D. Use the Startup Disk utility.

21. What is the RDISK command used for in a Windows NT 4.0 system?

    A. to create an ERD

    B. to restore a backup

    C. to restore a file system to a previous good condition

    D. to make a backup to a disk

22. What utility is used to create a backup copy of the Registry in Windows NT 4.0?

    A. RDISK

    B. BACKUP

    C. SYSBACK

    D. REGBACK

23. Which Windows XP utility is used to recover a malfunctioning system, if the Safe Mode, Last Known Good Configuration, and Recovery Console options have failed?

    A. the Emergency Repair Disk

    B. the Emergency Repair Process

    C. the System Restore function

    D. the Automated System Recovery tool

24. Which file loads the Windows 9x operating system into the system?

APPLY YOUR KNOWLEDGE

A. START.EXE

B. IO.SYS

C. WIN.COM

D. AUTOEXEC.BAT

Answers and Explanations

1. **A**. If the BIOS does not locate the boot record in one of the indicated drives, it will most likely display a `Non-System Disk` or `Disk Error` or `ROM BASIC Interpreter Not Found` message on the screen.

2. **C**. The files required to boot an MS-DOS system and their execution order are IO.SYS, MSDOS.SYS, CONFIG.SYS, COMMAND.COM, AUTOEXEC.BAT.

3. **C**. The special function keys available during MS-DOS bootup are F5 (also Left Shift key), which skips CONFIG.SYS and AUTOEXEC.BAT files; and F8, which proceeds through the CONFIG.SYS and AUTOEXEC.BAT files one step at a time.

4. **A**. The Windows 9x startup sequence can be summarized as follows:

    1. POST tests are run.
    2. The PnP capability is configured.
    3. The OS bootstrap looks for the MBR.
    4. The system loads IO.SYS.
    5. IO.SYS loads and executes CONFIG.SYS (if present).
    6. IO.SYS loads MSDOS.SYS.
    7. IO.SYS loads and executes COMMAND.COM.
    8. COMMAND.COM looks for and executes AUTOEXEC.BAT (if present).
    9. IO.SYS loads WIN.COM (the `Starting Windows` notice is displayed).
    10. The KERNEL32.DLL and KERNEL386.EXE files are executed.
    11. The GDI.EXE and GDI32.EXE files are executed.
    12. The USER.EXE and USER32.EXE files are executed.
    13. All fonts and other associated resources are loaded.
    14. The WIN.INI file values are checked.
    15. The Windows 9x shell and machine policies are loaded.
    16. The Windows desktop components are loaded.
    17. Windows 9x checks the Startup folder.

5. **C**. If the Normal option is selected, the system tries to restart normally, loading all its normal startup and Registry files. The Logged option also attempts to start the system in normal mode but keeps an error log file that contains the steps performed and their outcome. This text file (BOOTLOG.TXT) can be read with any text editor or printed out on a working system.

6. **C**. In Safe Mode, the minimal device drivers (keyboard, mouse, and standard-mode VGA drivers) are active to start the system; however, the CD-ROM drive is not active in Safe Mode.

APPLY YOUR KNOWLEDGE

7. **D**. The Startup menu allows you to access all the standard Safe Modes and others (including Normal, Logged, and standard Command Prompt Only modes), which can be accessed by pressing F8.

8. **A**. Standard Safe Mode can be accessed by pressing F5 when the `Starting Windows 9x` message is displayed onscreen.

9. **B**. You can use a question mark as a switch with the `WIN` command (that is, `WIN /?`) to show a listing of all the switches associated with the command. You can use these switches to start Windows with various portions of the operating system disabled.

10. **D**. The Windows NT/2000 boot sequence includes `NTLDR`, `BOOT.INI`, `NTDETECT.COM`, `NTOSKRNL.EXE`, and `HAL.DLL`.

11. **D**. The old Last Known Good Hardware Configuration settings are not replaced until a user actually logs on to the system.

12. **A**, **B**, **C**. The files required to boot to an MS-DOS command prompt are `IO.SYS`, `MSDOS.SYS`, and `COMMAND.COM`. The `CONFIG.SYS` and `AUTOEXEC.BAT` files are helpful in configuring a bootup but are not required to boot to a command prompt.

13. **B**, **C**. An emergency start disk is an essential tool for troubleshooting in Windows 9x. There are two methods for creating this disk. When you install the Windows 9x operating system, you are asked whether you want to create such a disk. For all other situations, you need to use the Add/Remove Programs tool in the Control Panel. When you open the Add/Remove Programs utility, just click on the Startup Disk tab and run the wizard. The file on this disk that is used to boot the system is `WIN.COM`.

14. **B**. The Windows 9x emergency start disk contains several useful utilities including FDISK, FORMAT, SYS, EDIT ATTRIB, REGEDIT, and SCANDISK.

15. **C**. The Windows NT/2000/XP boot sequence includes the files `NTLDR`, `BOOT.INI`, `NTDETECT.COM`, `NTOSKRNL.EXE`, and `HAL.DLL`. `NTBOOTDD.SYS` is used to enable SCSI disk drives for NT bootup.

16. **C**. You may want to include copies of any CD-ROM driver files on the emergency start disk. The CD-ROM driver (that is, `MSCDEX.EXE`) should be included to provide CD-ROM support for access to the utilities on the Windows distribution CD.

17. **C**. Files for the repair and update of the Windows 9x operating system can be extracted from the cabinet (`.CAB`) archive files on the Windows 98 distribution CD.

18. **D**. The Windows 98 emergency start disk features a new Extract command (`EXT.EXE`) to work with `EXTRACT.EXE`. The Extract command is used to pull necessary files from the cabinet (`.CAB`) files on the Windows 98 distribution CD.

19. **C**. To create Setup disks under Windows NT 4.0, you must install the Windows NT distribution CD in the system and type `WINNT /ox` at the command prompt.

APPLY YOUR KNOWLEDGE

20. **A**, **C**. The Windows 2000 Setup routine prompts you to create an ERD during the installation process. The ERD can also be created using the Windows 2000 Backup utility, which you can access by selecting Programs, Accessories, System Tools.

21. **A**. You can create a Windows NT ERD using the Repair Disk program (`RDISK.EXE`). To do so, select the Run option from the Start menu, enter the `CMD` command in the Run dialog box, and then type `RDISK` at the command prompt.

22. **A**. The `RDISK.EXE` utility, located in the `\Winnt\System32` folder, can be used to create a backup copy of the Windows NT Registry in the `\Winnt\Repair` folder.

23. **D**. The Windows XP ASR utility replaces the emergency repair disk from Windows 9x/2000. This utility is used specifically when these other efforts have been tried and have failed.

24. **C**. `IO.SYS` loads the `WIN.COM` file into RAM, which controls the loading and testing of the Windows 9x core components.

Challenge Solutions

1. There are two possible ways to start Windows 9x if it hangs up before reaching the desktop display. The first is to press the F8 key to access the Startup menu so that Safe Mode or another start-up method can be selected. The second method is to start the system with a boot disk and use the `WIN/D` command to start Windows. Either of these options should permit you to begin troubleshooting the Windows environment.

2. You must load the `MSCDEX.EXE` driver for the CD-ROM drive on the system. This enables the CD-ROM to operate under Safe Mode (remember that in Safe Mode, only the keyboard, video display, and mouse drivers are loaded into the system).

3. You must insert the Windows 98 emergency start disk and extract the `SCANDISK.EXE` file from the `.CAB` file (`EDB.CAB`) using the Extract command.

4. You should tell your associate to select the Run option from the desktop's Start menu, enter the `CMD` command in the Run dialog box, and then type `RDISK` at the command prompt.

APPLY YOUR KNOWLEDGE

Suggested Readings and Resources

1. POST

   http://www.yale.edu/pclt/BOOT/POST.HTM

2. Plug-and-Play

   http://www.pcmech.com/show/cards/143/

3. Plug-and-Play Guide

   www.pcguide.com/ref/mbsys/res/pnp.htm

4. DOS Bootup

   http://www.pccomputernotes.com/operating_systems/dosboot.htm

5. DOS System Files

   http://pclt.cis.yale.edu/pclt/BOOT/DOS.HTM

6. Windows 95 Startup

   http://support.microsoft.com/directory/article.asp?ID=KB;EN-US;Q174018&FR=1&LNG=ENG&SA=PER&

7. Windows 98 Startup

   http://www.geocities.com/~budallen/98start.html

8. Kernel32.dll

   http://webopedia.internet.com/TERM/k/kernel32_dll.html

9. Windows 9x Safe Mode

   http://service4.symantec.com/SUPPORT/tsgeninfo.nsf/docid/1999101916343139

10. Windows 95/98 WIN.COM Command-Line Switches

    http://support.microsoft.com/default.aspx?scid=kb;EN-US;q142544

11. Windows 2000 Startup

    http://www.microsoft.com/windows2000/techinfo/reskit/en/ProRK/prbd_std_nfkh.htm

12. Windows 9x Emergency Start Disk

    http://www.pcworld.com/howto/article/0,aid,44202,00.asp

13. Windows NT/2000 Emergency Repair Disk (ERD)

    http://is-it-true.org/nt/nt2000/atips/atips32.shtml