CHAPTER 9

LOCAL AREA NETWORK OPERATING SYSTEMS AND REMOTE ACCESS

Concepts Reinforced

- OSI model
- Protocols and standards
- Network architectures
- Top-down model
- Hardware/software compatibility

Concepts Introduced

- Network operating system functionality
- Peer-to-peer network operating systems
- Network technology analysis
- Client network operating systems
- Remote access
- Mobile computing
- Functional network analysis
- Client/server network operating systems
- Network operating systems architectures
- Server network operating systems
- Remote control
- Remote access security

OBJECTIVES

After mastering the material in this chapter you should:

1. Understand the compatibility issues involved with implementing LAN software.
2. Understand the basics of network operating system functionality.
3. Understand the important differences between peer-to-peer and client/server network operating systems architectures.
4. Understand the emerging role of the client network operating system and the universal client.
5. Understand how to analyze functional networking requirements and match those requirements to available technology.
INTRODUCTION

Network operating systems, like most other aspects of data communications, are undergoing tremendous change. As a result, before examining the operational characteristics of a particular network operating system, it is important to gain an overall perspective of network operating systems in general. In particular, network operating systems architectures are in a state of transition from closed environments in which only clients and servers running the same network operating system could interact, to open environments in which universal clients are able to inter-operate with servers running any network operating system.

In this chapter network operating system functionality is examined for both client and server network operating systems. This functionality is representative of current network operating systems in general rather than any particular product.

With the evolution of portable computers and the Internet users need to gain access to an organization’s data from a variety of locations other than the traditional office setting. One of the most important things to understand about such remote access is the relatively limited bandwidth of the wide area network links that individuals will use to connect to the main office information resources. Although the goal of remote access may be to offer transparent remote connectivity, decreases in bandwidth by a factor of 100 on WAN links as compared to LAN links cannot be ignored.

The overall goal of the second half of this chapter is to outline a methodology for the proper design of remote access solutions based on a thorough understanding of user needs, network architecture alternatives, and available technology.

NETWORK OPERATING SYSTEMS OVERVIEW

Traditionally, there were two major product categories of network operating systems: peer-to-peer and client/server. In a peer-to-peer network operating system, individual workstations can be configured as a service requester (client), a service provider (server), or both. The terms client and server in this case describe the workstation’s functional role in the network. The installed network operating system is still considered a peer-to-peer network operating system, because all workstations in the network use the same networking software. Designed as a low cost, workgroup solution, peer-to-peer network operating systems lacked the ability to offer centralized authentication and authorization and suffered from exponential performance decreases as the number of users increased. As a result, peer-to-peer network operating systems were often characterized as lacking scalability.

In contrast to the homogeneous, peer-to-peer software environment, traditional client/server network operating systems require two distinct software products for client and server computers. The specialized client software required less memory and disk space, and was less expensive than the more complicated and expensive server software. The client software was made to interact with the corresponding server software. As a result, although traditional client/server network operating systems overcame the scalability limitation of peer-to-peer network operating systems, they did not necessarily overcome the interoperability limitation. Functionally, client/server network operating systems offered faster, more reliable performance than peer-to-peer LANs and well as improved administration, scalability, and security.
Functional Requirements of Today’s Network Operating Systems

Although traditional peer-to-peer and client/server network operating systems successfully met the functional requirements for workgroup and departmental computing, as these departmental LANs needed to be integrated into a single, cohesive, interoperable, enterprise-wide information system, the limitations of these traditional NOS (network operating system) architectures became evident.

In order to understand the architectural specifications of today’s network operating systems, it is first necessary to understand the functional requirements that these network operating systems must deliver. In taking a top-down approach to network operating system requirements analysis, one might ask, “What are users of an enterprise-wide information system demanding of a network operating system in terms of services?” The answer to this question lies in the application layer of the top-down model. Given that it is distributed applications that will enable enterprise-wide productivity and decision making, the underlying network operating systems must support these distributed applications by supplying the message services and global directory services required to execute these applications in an enterprise-wide, multiple server environment.

Figure 9-1 illustrates these functional requirements and contrasts them with the requirements traditionally demanded of client/server and peer-to-peer network operating systems.

As illustrated in Figure 9-1, the new or emerging demands being put on network operating systems are application services, directory services, and integration and migration services. In order to successfully meet these functional requirements, network operating system architectures have shifted from integrated, single-vendor client/server network operating systems to independent, distinct, multivendor, client and server network operating systems. The functional characteristics of these distinct client and server network operating systems are described in detail later in this chapter. Figure 9-2 illustrates this architectural shift in network operating system development.
Client Network Operating Systems: The Universal Client

Client network operating systems, as illustrated in Figure 9-2, integrate traditional operating system functionality with highly configurable networking features to enable communication with a variety of network operating system servers. The client workstation’s ability to interoperate transparently with a number of different network operating system servers without the need for additional products or configurations breaks the traditional hard linkage between client and server NOS. This ability is commonly referred to as universal client capability.

Server Network Operating Systems

Because the client and server platforms have been de-coupled, server network operating systems can be selected based on their performance characteristics for a given function. For example, Windows servers are often employed as file and print servers while UNIX servers are most likely to be employed as database and application servers. Because the universal client has the ability to communicate with any server, and the server has the ability to communicate with any client, the choice of server network operating system can be based on optimizing functional performance rather than whether the system simply provides interoperability.
CLIENT NETWORK OPERATING SYSTEM FUNCTIONALITY

Having gained an understanding of the new architectural arrangement of network operating systems consisting of distinct, interoperable, multivendor, client and server network operating systems, the functional aspects of client network operating systems categories can be examined.

Client network operating systems such as Microsoft Windows and the Macintosh OS offer three major categories of functionality:

1. Operating system capabilities
2. Peer-to-peer networking capabilities
3. Client software for communicating with various network operating systems

The logical relationship of these three distinct yet complementary categories of functionality is illustrated in Figure 9-3. Figure 9-3 also points out potential areas for compatibility and protocol consideration where the various software and hardware layers interface.

The following sections cover the importance of each of these three functional categories to the overall network operating system, as well as key implementation differences between technologies. Following this overview, a network analyst should be able to construct a logical network design listing the functionality required to meet the business objectives.

This logical network design is then used as an evaluation mechanism for selecting available technologies. Logical network design functionality can be compared to available technology’s delivered functionality in a technology analysis grid such as Figure 9-7 (“Client Network Operating System Technology Analysis Grid”). As stated in previous chapters, the advantage to employing a technology analysis grid in such an endeavor is that it assures that purchase decisions or recommendations are made based on facts rather than creative packaging or effective marketing.
Operating System Capabilities

The following operating systems characteristics are listed and briefly explained here from the perspective of each characteristic’s importance to overall network operating system performance.

- **32-bit operating system.** 32-bit operating systems will allow more sophisticated and higher-performance 32-bit applications to execute more quickly. Although 64-bit hardware is coming to market, operating systems have yet to move to a 64-bit platform.

- **Pre-emptive multitasking.** Pre-emptive multitasking prevents misbehaving programs from monopolizing systems resources at the expense of the performance of other applications.

- **Protected memory space.** Protected memory space prevents application programs from accidentally writing into each other’s or the operating system’s memory space thereby causing general protection faults and/or system crashes.

- **Support for symmetrical multiprocessing.** SMP support is especially important for server network operating systems due to the processing load imposed by multiple simultaneous requests for services from clients. Some high-powered client applications such as 3-D modeling or simulation software may warrant SMP support on client platforms as well.

- **Multithreading.** Multithreaded applications are only able to achieve performance increases if they are executed by an operating system that supports multithreaded applications, allowing more than one sub-process to execute simultaneously.

Application Program Support  A very important aspect of any migration plan to a new client network operating system is the extent of support for backward compatibility is terms of application support, also known as legacy application support. It should stand to reason that most companies cannot afford to replace or re-write all of their application software in order to upgrade to a new client network operating system.

Although 32-bit client network operating systems are desirable and most current network-based applications are 32-bit, many custom software solutions are still 16-bit. In addition, many of these 16-bit application programs bypass supported API calls and commands in favor of directly addressing hardware devices. Initially done in the interest of increasing performance, these applications significantly limit multitasking and interoperability. Programs or subroutines that write directly to computer hardware are sometimes referred to as employing real-mode device drivers.

Many 32-bit network operating systems do not allow application programs to address or control hardware directly in the interest of security and protecting applications from using each other’s assigned memory spaces and causing system crashes. Instead, these more secure 32-bit operating systems control access to hardware and certain system services via virtual device drivers, otherwise known as VxDs. Windows XP is a good example of a 32-bit network operating system that prevents direct hardware addressing. As a result, many 16-bit applications, particularly highly graphical computer games, will not execute over the Windows XP operating system. On the other hand, Windows XP is extremely stable.
Another issue concerning the execution of 16-bit applications is whether those applications execute in a shared memory address space, sometimes referred to as a 16-bit sub-system. If this is the case, then a single misbehaving 16-bit application can crash the 16-bit subsystem and all other executing 16-bit applications. Some 32-bit operating systems allow each 16-bit application to execute in its own protected memory execution area.

When it comes to 32-bit applications, client network operating systems may execute these applications in their own address space, otherwise known as protected memory mode. However, all of these protected-mode 32-bit applications may execute over a single 32-bit sub-system, in which case a single misbehaving 32-bit application can crash the entire 32-bit subsystem and all other associated 32-bit applications.

Whether or not an application is executable over a particular network operating system is dependent upon whether that application issues commands and requests for network-based services in a predetermined format defined by the network operating system’s application program interface (API). Each network operating system has a unique API or variation. For example, Microsoft 2000/XP and Windows 9x both support variations of the Win32 API.

Some client network operating systems, such as Windows 2000/XP, have the ability to support multiple APIs and multiple different operating system subsystems, sometimes known as virtual machines. This feature allows applications written for a variety of operating systems such as OS/2, DOS, or POSIX, to all execute over a single client network operating system.

Figure 9-4 illustrates some of the concepts of Application Program Support by Client Network Operating Systems.

Plug-n-Play Traditionally, one of the largest problems with installing new devices into a computer was configuring the hardware’s resource usage. Plug-n-play (PnP), included in most modern client network operating systems, is designed to free users from having to understand and worry about such things as IRQs (Interrupt Requests), DMA (Direct Memory Access) channels, memory addresses, COM ports, and manually editing configuration files to add a device to their computer.

Although the goal of completely automatic hardware configuration has not been fully realized, definite progress has been made. Ideally, PnP functionality will automatically detect the addition or removal of PnP devices, configure them so that they do not conflict with other devices, and automatically load necessary drivers to enable the particular device.

Compatibility issues are important to the achievement of full PnP functionality. To be plug-n-play compliant, the following are required:

- A PnP BIOS (Basic Input Output System) is required to interface directly to both PnP and non-PnP compliant hardware.
- PnP capabilities must be supported by the operating system through interaction with the PnP BIOS.
- The devices installed must be PnP compliant. This basically means that the manufacturers of these devices must add some additional software and processing power so that these devices can converse transparently with the PnP operating system and BIOS. In some cases, PnP compliant device drivers may also be required.
Devices are detected by the client operating system through an assistant agent program, sometimes referred to as a hardware wizard, which walks the user through the configuration routine. Such programs are often capable of detecting and displaying IRQs and DMA addresses used by other devices, allowing users to accept supplied default answers in this semi-automatic configuration scenario.

**Peer-to-Peer Networking Capabilities**

Most current network client operating systems also include peer-to-peer networking capability. These features allow each client to interact at a basic level with other clients. By utilizing these features, many small businesses can avoid the expense of implementing a large-scale server environment.
**File and Printer Sharing** Perhaps the most basic peer-to-peer network function is file and printer sharing. In many cases, other resources such as CD-ROM drives and fax modems can also be shared. Network operating systems supporting peer-to-peer networking can vary widely in terms of file access security features. The level at which access can be controlled (disk, directory, or file level) is sometimes referred to as the **granularity** of the access control scheme. Access is controlled on a per-user or per-group basis. Sophistication of the printer management facility can also vary from one client network operating system to another.

**Workgroup Applications** Striving to find new ways to differentiate themselves from the competition, client network operating systems are usually offered with bundled workgroup application software such as these:

- Terminal emulation
- Calculator
- Clock
- Games
- Paintbrush
- Sound recorder
- Remote access software
- CD player
- Backup
- Chat
- Phone dialer
- Performance and network monitors
- Diagnostic software
- Screen savers
- Web browsers
- Internet telephony
- Instant messenger clients
- Fax access software

**Managerial Perspective**

*JUST BECAUSE SOMETHING IS FREE DOES NOT NECESSARILY MAKE IT VALUABLE*

The network operating system that offers the greatest number of workgroup applications is not necessarily the best or most appropriate choice. Although free application software is nice, priority should be given to client network operating systems characteristics:

- Application program support and operating system characteristics
- Networking capabilities
- Flexibility and ease of installation and use in acting as a client to a variety of different server network operating systems

Client network operating systems that are able to connect to many different server operating systems are sometimes referred to as a universal client. In support of multivendor, multiplatform, distributed information systems, this is probably the most important evaluation criteria when selecting a client network operating system.

**Client Networking Capabilities**

As illustrated architecturally in Figure 9-5, there are three distinct elements of client network functionality in addition to the previously mentioned required application support capabilities. In some cases, more than one alternative is offered for each of the following elements:
- **Client software and network drivers** allow a particular client to communicate with a compatible server.

- **Network transport protocols** package and transport messages between clients and servers. These protocols correspond to the network and transport layers of the OSI model.

- **Network redirectors** trap API (application program interface) calls and process them appropriately. Redirectors are concerned with providing file system related services in support of application programs.

More than one protocol may be provided in a given client network operating system for each of the three network protocol categories. Figure 9-6 displays the protocol stacks Windows ’9x, Windows XP, and Linux. Rather than organize protocols according to the OSI Network Reference Model, Figure 9-6 divides the protocols into layers according to networking functionality.

**Connecting Clients to Multiple Servers**  In most client network operating systems, the combination of these three elements of network functionality allows client platforms to automatically find and connect to servers. For example, a properly configured Windows client will be able to automatically display network connections and connect to Windows and NetWare servers that are physically reachable and
to which the client has been assigned access privileges. The client software does not have to be preconfigured with any information about these servers. The server discovery and access is all handled transparently by the client network operating system.

In addition to network operating system client software for specific server network operating systems such as NetWare and Windows NT Server, specialized application-oriented client software is often also included in client network operating systems. Examples of such network applications include the following:

- Web browsers
- FTP (file transfer protocol) client software
- E-mail client software
- Scheduling systems client software

<table>
<thead>
<tr>
<th>Application Support</th>
<th>Windows XP</th>
<th>Windows '9x</th>
<th>Linux</th>
</tr>
</thead>
<tbody>
<tr>
<td>WIN32 API</td>
<td>32-bit and some 16-bit Windows applications supported</td>
<td>WIN32 API</td>
<td>32-bit and most 16-bit Windows applications supported</td>
</tr>
<tr>
<td>Application Redirectors and File Systems</td>
<td>NCP (Netware Core Protocol Redirector (Novell))</td>
<td>SMB (Server Message Block Redirector (Microsoft))</td>
<td>FAT (File Allocation Table for DOS/Windows)</td>
</tr>
<tr>
<td>MAC Sublayer Specifications</td>
<td>NDIS (Network Data-Link Interface Specification (Microsoft/3Com))</td>
<td>NDIS (Network Data-Link Interface Specification (Microsoft/3Com))</td>
<td>ODI (Open Data-Link Interface (Novell))</td>
</tr>
</tbody>
</table>

Figure 9-6  Supported Protocol Stacks for Major Client Network Operating Systems
In the case of the e-mail and scheduling clients, maximum benefit is attained only when compatible e-mail and scheduling application servers are available. The client portion is merely the front-end to a back-end application engine executing in some other network-accessible location. Most e-mail clients do support the POP3 and SMTP protocols required to connect to generic Internet mail servers.

**Mobile Computing Synchronization**  As mobile computing on notebook computers has grown exponentially, a need to synchronize versions of files on laptops and desktop workstations became quickly apparent. Such *file synchronization software* was initially available as a stand-alone product or included as a feature on remote access or file transfer packages. Also known as *version control software* or *directory synchronization software*, this valuable software is now often included as a standard or optional feature in client network operating systems. Laptop synchronization should happen automatically when the laptop computer is docked in its docking station. E-mail clients and scheduling system client software should automatically synchronize with the LAN-attached e-mail and scheduling application servers.

Some of the important functional characteristics or differences among laptop synchronization options include:

- **Copy by date option.** Files and directories can be selectively synchronized by selected data range.
- **Bidirectional option.** File synchronization can occur just from laptop to desktop, desktop to laptop, or both (bi-directional).
- **Cloning option.** This guarantees that the contents of a directory on one system exactly matches the contents of the same directory on another system.
- **Refresh option.** This copies only newer versions of files that are already located on both systems from one system to another.
- **Delta file synchronization.** This is perhaps the most significant file synchronization option in terms of its potential impact on reducing required bandwidth and file transfer time to accomplish the synchronization. Rather than sending entire files across the dial-up or LAN link, delta file synchronization only transfers the changes to those files.

**CLIENT NETWORK OPERATING SYSTEM TECHNOLOGY ANALYSIS**

Figure 9-7 is a technology analysis grid comparing key architectural and functional characteristics of Windows ’9x, Windows XP, and Linux.

This grid is included as an example of how technology analysis grids can be used to effectively map required networking functional requirements to available technology solutions in an objective manner. This technology analysis grid is not meant to be absolutely authoritative or all-inclusive. Its primary purpose is to provide a concrete example of the type of analysis tool used in a professional, top-down, network analysis and design methodology. It is expected that network analysts will create new technology analysis grids for each networking analysis opportunity based on their own networking functional requirements and the latest technology specifications available from buyer’s guides or product reviews.
The client network operating system technology analysis grid is divided into the following major sections:

- Hardware/platform-related characteristics
- Operating system capabilities
- Peer-to-peer networking capabilities
- Client networking capabilities

| Client Network Operating System Technology Analysis Grid |
|---------------------------------|-----------------|-----------------|-----------------|
| **Category** | **Windows 9x** | **Windows XP** | **Linux**       |
| **Hardware & Platform** | | | |
| Required-Recommended Memory | 32–64 MB | 64–128 MB | 32–128 MB |
| 16 or 32 bit | 32 bit | 32 bit | 32 bit |
| **Operating System Capabilities** | | | |
| Preemptive Multitasking | yes | yes | yes |
| Supports SMP (professional version) | no | yes | yes |
| Protected Memory Program Execution | yes | yes | yes |
| Multithreading | yes | yes | yes |
| Runs 32-bit apps | yes | yes | yes |
| Runs 16-bit apps | yes | Some; won’t support real mode drivers | no |
| **Peer-to-Peer Networking** | | | |
| File & Printer Sharing | yes | yes | yes |
| Workgroup Applications | yes | yes | yes (most distributions) |
| **Client Networking** | | | |
| Network Clients | Windows Servers, NetWare | Windows, NetWare, UNIX (NFS) | Windows (SAMBA), UNIX (NFS) |
| Network Transport Protocols | NetBEUI, TCP/IP, IPX/SPX | NetBEUI, TCP/IP, IPX/SPX | TCP/IP, IPX/SPX |
| Remote Access | yes | yes | yes (most distributions) |
| Laptop Synchronization | yes | yes | no |

*Figure 9-7  Client Network Operating System Technology Analysis Grid*
SERVER NETWORK OPERATING SYSTEM FUNCTIONALITY

Changing Role of the Server Network Operating System

Traditionally, file and printer sharing services were the primary required functionality of server-based network operating systems. However, as client/server information systems have boomed in popularity, application services have become one of the most important criteria in server network operating system selection. The distributed applications of the client server model require distinct client and server portions applications to interact in order to perform the required task as efficiently as possible. The server network operating system is responsible for not only executing the back-end portion of the application, but also supplying the messaging and communications services that enable interoperability between distributed clients and servers. Figure 9-8 illustrates the evolving role of the server network operating system from an architectural perspective.

The remainder of this section will focus on server network operating system functionality—specifically, those aspects of functionality that are most important to the support of distributed applications and their associated distributed clients and users.
Novell NetWare and Microsoft Windows Server are the predominate server network operating systems. When comparing these two network operating systems, it is important to note the historical strong points of each. NetWare has traditionally been stronger in file and print services than in the area of application services. Microsoft Windows Server has traditionally been stronger in terms of application services than file and print services. With the release of the latest versions of each product, both are rapidly making progress at improving on their weaknesses.

Various flavors of UNIX, combined with TCP/IP as a network protocol and NFS as a file system, have also been a popular choice of an applications server platform. However, this combination of operating system, network protocols, and file system is not as integrated or feature-rich as either NetWare or Windows NT Server and probably does not deserve the label of “next-generation” NOS. To resolve issues with NFS, newer UNIX implementations have co-opted the SMB protocol from Windows through the use of SAMBA to allow them to appear as a Windows NT Server on the network. The combination of UNIX and SAMBA makes an attractive NOS solution.

**Directory Services**

Network operating systems depend on some sort of naming service or directory in which to store information about users and system resources such as disks, servers, and printers. Traditionally each individual server maintained this list separately. However, as distributed client/server systems became more prevalent a means of sharing this information was required. There are two approaches to accomplishing this: domains and directory services.

**Directory Services vs. Domains** Domains and directory services differ primarily in the organization of information concerning network users and resources. Directory services organize all network user and resource data into a single hierarchical database, providing a single point of user and resource management. The hierarchical database is based on a hierarchical tree structure. All servers that participate in the directory are part of the global hierarchy and can see all other parts of the network. In this sense, the hierarchical directory database is merely a reflection of the hierarchical network itself. Examples of directory services include Novell’s Netware Directory Service (NDS) and Microsoft’s Active Directory (AD).

The directory service database is often **distributed**; different portions of the data are physically stored on different servers connected via the network. In addition, the directory service database is often **replicated** among multiple servers for redundancy and fault tolerance. In terms of a logical view of the network, directory services provide a view of a single, enterprise network.

In contrast, **domains** see the network as a series of linked subdivisions. Domain’s associate network users and resources with a special server known as a **Domain Controller**. Each domain’s directory must be individually established and maintained. Domains can be individually maintained and controlled in terms of how much of other domains can be seen. The key weakness with the domain model is that there is limited granularity: You are in the domain or you are not. There is no hierarchy within the domain. Examples of domain solutions include Microsoft Windows NT and SAMBA running on UNIX.

Directory services can also vary in what types of information are stored in the directory services database. In some cases, all users and network resources are...
considered **network objects** with information concerning them stored in a single database, arranged by object type. Object attributes can be modified and new network objects can be defined. In other cases, network users and network resources are kept in separate databases. Frequently, separate databases are maintained for network user account information and e-mail user account information.

### DIRECTORY SERVICES COMMUNICATION

In a directory service implementation, a remote server performs a lookup in the directory database to authenticate the user’s right to the requested service. This database lookup is repeated for every request for service from remote users. Recalling that the database is distributed, the physical location of the server which contains the rights information of the requesting user may be located anywhere in the hierarchical distributed network.

In the case of a domain solution such as SAMBA, the remote or foreign server receives the user authentication from the user’s domain controller (local server) in a process known as **Inter-domain Trust (IT)**. By having servers act on behalf of their local users when verifying authenticity with remote and foreign servers, every user ID does not have to be entered and maintained in every domain’s directory service. In addition, once the interdomain trust has been established for a particular user, the remote domain server does not repeat the request for authentication.

The current trend is toward directory services and away from domain solutions. While each vendor’s directory services solution is proprietary, all are loosely based on the OSI **X.500** directory service standard. A subset of the X.500 standard, known as the **Lightweight Directory Access Protocol or LDAP** and standardized by both OSI and the IETF, provides a means of achieving some degree of interoperability between various vendors directory services offerings. LDAP runs over TCP/IP on ports 389 and 636. Any client that supports LDAP can access information in any LDAP compliant directory service. All major vendors support LDAP in their various directory services offerings.

### Application Services

Recalling that the primary objective of the next-generation server NOS is to provide high-availability, high-performance application services, the most important NOS characteristic is the ability to support symmetrical multiprocessing. As numbers of users and sophistication of application programs continue to increase, the only real solution is for the application to be able to utilize more processing power simultaneously. Not all server network operating systems support symmetrical multiprocessing and those that do might vary in the maximum number of processors supported. Other server network operating system characteristics that are essential to optimization of application program performance:

- Preemptive multitasking
- 32-bit execution
- Multithreaded application support
- Program execution in protected memory space
**File Services** Applications programs are stored in a particular file system format. In addition, when these application programs execute, they may request additional services from the resident file system via API calls. Server network operating systems vary in the types and number of supported file systems. Some network operating systems can have multiple partitions on a disk drive supporting multiple file systems. Figure 9-9 lists file systems supported by the various server network operating systems in the marketplace.

Other file services offered by some server network operating systems include file compression utilities and **data migration** utilities that manage the migration of data among different types of storage devices as part of a comprehensive hierarchical storage management (HSM) program.

**Application Integration** Application integration refers to the extent to which applications program are able to integrate or take advantage of the capabilities of the operating system in order to optimize application program performance. Successful applications integration with operating system can yield both increased convenience and performance.

From a convenience standpoint:

- Does the application integrate with the operating system’s security system, allowing single-user accounts, or must two separate security databases be maintained?
- Does the application integrate with the operating system’s monitoring capabilities, allowing it to be monitored from within the operating system?
- Can the application be configured and maintained from within the operating system’s control panel or setup subsystem?

From a performance standpoint:

- Can the application take advantage of the multithreaded capabilities of the operating system?
- Can the application automatically detect the presence of multiple processors and respond accordingly?

<table>
<thead>
<tr>
<th>File System Name</th>
<th>Associated Network Operating System</th>
</tr>
</thead>
<tbody>
<tr>
<td>EFS</td>
<td>UNIX/Linux</td>
</tr>
<tr>
<td>EFS2</td>
<td>UNIX/Linux</td>
</tr>
<tr>
<td>FAT 32—File Allocation Table/32</td>
<td>Windows 9x, 2000, XP, and Linux</td>
</tr>
<tr>
<td>NetWare File System</td>
<td>NetWare 3 and 4</td>
</tr>
<tr>
<td>NetWare Storage System</td>
<td>NetWare 5 and 6</td>
</tr>
<tr>
<td>NTFS—NT File System</td>
<td>Windows NT, 2000, XP, server 2003, and Linux</td>
</tr>
<tr>
<td>VFS</td>
<td>UNIX/Linux</td>
</tr>
</tbody>
</table>

*Figure 9-9* File Systems and Associated Server Network Operating Systems
• Can the application use the multitasking capabilities of the operating system, or does it supply its own multitasking environment?
• How easily, and to what extent, can adjustments be made to the operating system in order to optimize the performance of the application?

**Networking and Connectivity Services**

**Network Client Support** In addition to the client network operating systems that were previously reviewed, server network operating systems may also have to communicate with client platforms with the following operating systems installed:

- Windows 9x
- Windows NT/XP
- Macintosh
- UNIX

Many of these operating systems provide built-in client software that will allow for it to connect to the server. However, often this built-in software is rather limiting in terms of its overall functionality and ability to integrate cohesively. It is usually best to ensure that the latest client software available from the server vendor is installed on the client.

**Network Protocol Support** One of the key questions concerning network protocols and server network operating systems is how many different network protocols are supported and how many network protocols can be supported simultaneously? Although the standardization on TCP/IP as the network protocol of choice in recent years has made this process easier, it is still important to ensure that the systems can support all protocols that may be required.

Related to the ability of a server network operating system to simultaneously support multiple protocols is the ability of a server network operating system to support multiple network interface cards. If a single NIC is the bottleneck to network communications, additional NICs can be added, provided the NOS and computer bus supports them.

**Remote Access and Gateway Services** Just as client network operating systems supplied the client portion of a remote access communication, server network operating systems can supply the server side of remote access communication. These remote access servers may be included with the server NOS or may be available for an additional fee. It is important that these remote access servers be tightly integrated into the server network operating system to assure reliable performance, full functionality as offered to locally connected users, and tight security. Remote access solutions are detailed later in this chapter.

**Management & Administration Services**

**Installation, Configuration and Administration** Reviews of server network operating systems consistently list auto-detection and configuration of installed controllers,
interface cards and peripherals as the most important installation-related feature. The ability of a server network operating system to automatically configure a controller, adapter, or peripheral is dependent on the network operating system possessing a compatible driver for that device. It should stand to reason that the greater the number of drivers supported by a given network operating system, the greater the probability that auto-configuration will be successful.

In order to appreciate the differences in ease of administration offered by server network operating systems, it is important to consider enterprise network serving hundreds, if not thousands, of users. Simple items that are merely a nuisance on a smaller network can easily become major issues in such a large implementation. With this scenario in mind, there are some pertinent questions:

- How many steps are involved in creating a new user account?
- What is involved in giving a user access to remote servers?
- How easily can a user profile be copied and used as a template to automatically generate other user profiles? (This feature is particularly important in academic settings where user profiles must be constantly generated in large numbers.)
- What tools are available to assist in managing multiple servers simultaneously?

Server network operating systems can vary widely in the sophistication of the performance monitoring software included or available as an add-on. Ideally, the monitoring software should offer the ability to set thresholds for multiple system performance parameters. If these thresholds are exceeded, alerts or alarms should notify network management personnel of the problem, and offer advice as to possible diagnoses or solutions. Event logging and audit trails are often included as part of the performance monitoring package.

In multiple-server environments, it is particularly important that all servers can be monitored and managed from a single management console. Desktop and server management software offers capabilities beyond the monitoring software included in server network operating systems. For example, performance statistics are often gathered and stored in databases known as MIBs (Management Information Base). In addition, this performance management information can be communicated to Enterprise Management Systems such as HP OpenView, Computer Associates Unicenter, or Tivoli’s Management Framework using SNMP (Simple Network Management Protocol). In addition to these enterprise network management platforms, both Microsoft and Novell offer tools designed to ease system administration across large network installations such as Microsoft System Management Server (SMS) and Novell ManageWise. Network Management is covered in more detail in chapter 11.

Integration and Migration Migration features are aimed at easing the transition from one server NOS to another. Key among the migration concerns is the conversion of the directory services information. Utilities are available from third-party software vendors as well as from the NOS vendor’s themselves to help automate directory data conversion. Integration refers to the transition period of time in the migration process when both network operating systems are running simultaneously and interacting to some degree.

Monitoring As more mission-critical applications are deployed on LAN-attached servers, server operating systems must offer more sophisticated management tools in order to manage those applications effectively. Monitoring ability is essential in
determining where potential performance bottlenecks might occur and to react accordingly. Server attributes that should capable of being monitored and logged include the following:

- Processor utilization
- Network I/O
- Disk I/O
- Memory usage including L2 cache
- Individual application performance and system impact
- Process and thread performance

The monitor tool should be able to display data in a variety of ways:

- As a graph
- As a report
- As alarms or alerts if pre-set thresholds are crossed

A strong and flexible alert system is essential to keeping applications running and users happy. Some alert systems have the ability to dial particular pagers for particular alerts and can forward system status information to that pager as well.

A monitoring tool should support multiple open monitoring windows simultaneously so that multiple attributes or applications can be observed. The monitoring or management tool should be open and support industry standard management protocols and APIs so that application-specific management tools can be easily integrated into the overall operating system monitor.

**Performance**

In order to take advantage of the increased processing powers of multiple processors, server operating systems must be specially written to operate in a symmetrical multiprocessing (SMP) environment. Server operating systems can differ as to the maximum number of processors that they can support.

RISC-based UNIX systems such as Solaris and AIX have been offering SMP capabilities for many more years than Intel-based operating systems. As a result, UNIX-based server operating systems tend to be more stable, with larger number of processors.

Clustering is another performance-enhancing solution now available for LAN-based servers as more powerful and performance-hungry applications migrate from the mainframe environment to client/server architectures. By truly distributing applications on a thread level, as well as associated data over multiple CPUs physically located on multiple machines, clustering-capable operating systems can truly harness all of the available computing power in a client/server environment.

**Security**

Security will be covered in detail in chapter 12, but a review of basic security concepts is appropriate here. Overall security features fall into three broad categories:
Authentication

Authentication is concerned with determining which user is attempting to access the system. There are two key components to authentication: identification and proof of identification. In most network operating system environments, identification is provided by a UserID and proof of identification is provided by a password. Collectively, the UserID and password are known as a set of Authentication Credentials. In the login process, the server NOS checks the supplied authentication credentials against the directory service to determine if the credentials are valid. If so, the user is allowed to log into the system.

Most server network operating systems have the facility to support a guest user that is not authenticated. Although such a user usually is assigned minimal access rights to network resources, some situations, such as Internet publishing, require that a nonauthenticated user be given access to files and resources on a network server. When browsing the Web, a user typically doesn’t need to log onto each server they wish to access.

Authorization

Authorization is the process of controls access rights to network resources. Access Control Lists (ACLs) are the most commonly used authorization technique for local area network operating systems. In an ACL-based authorization scheme, a list of users and groups is attached to each network resource along with their permitted access level. When a user attempts to access a network resource, the server checks the supplied authentication credentials against the access control list. If the credentials support the desired access level, the action is completed. If the credentials do not support the desired access level, the action is prohibited and an error is displayed to the user. A second authorization method is Kerberos, a multiserver authorization method that provides increased security for large client/server applications. Although traditionally used only in mainframe type applications, Kerberos is rapidly being integrated into PC local area network operating systems to increase the level of available security as more high end applications are ported to these platforms.

In addition to individual user accounts, it is also possible to create user group accounts. A group account is a place holder that allows for a single change in user permissions or rights to affect a multiple users at a time. By assigning all authorization rights to groups rather than users, it is possible to greatly reduce the amount of effort required to administrate a system while increasing consistency and alleviating potential security holes. When creating user accounts, considerable time can be saved if UserIDs can be created or modified using a template or group name instead of having to answer numerous questions for every individual UserID.

Encryption

Encryption is the process of “scrambling” data before data are sent across a network and “unscrambling” it at the destination. Encrypting dataprotects the data from anyone who may make a copy along the way. There are multiple methods of data encryption in the marketplace. The two most common are DES and RSA public key encryption. DES is a single key system, whereby a single key is used to encrypt and decrypt the message. In RSA public key encryption a combination of private and public keys are used to encrypt and decrypt the message, effectively eliminating potential problems with the transmission of a single key from the source to the destination.
A network analyst’s job is to always seek out the latest information that the industry has to offer before making recommendations for purchases which could have a significant bearing on the company’s prosperity as well as personal job security. The following “Server Networking Operating System Technology Analysis Grid” (Figure 9-10) is

<table>
<thead>
<tr>
<th>Server Network Operating System Characteristic</th>
<th>Windows Server</th>
<th>NetWare</th>
<th>UNIX</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware/Platform</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Required Memory</td>
<td>128–512 MB</td>
<td>128–512 MB</td>
<td>64 MB +</td>
</tr>
<tr>
<td>CPUs</td>
<td>Intel</td>
<td>Intel</td>
<td>Intel, Sparc, Power PC, DEC/Compaq Alpha, MIPS, etc.</td>
</tr>
<tr>
<td>Symmetrical Multiprocessing</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Pre-emptive multitasking</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Multithreading</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Protected memory app execution</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td><strong>Installation &amp; Configuration</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Automatic detection &amp; configuration of adapters &amp; peripherals</td>
<td>yes</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Requires a separate administrator console</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td><strong>Networking &amp; Connectivity</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Clients supported</td>
<td>Windows, Macintosh, UNIX</td>
<td>Windows, Macintosh, UNIX</td>
<td>Windows, Macintosh, UNIX</td>
</tr>
<tr>
<td>Network Protocols supported</td>
<td>TCP/IP, IPX/SPX, NetBEUI, Appletalk, TCP/IP encapsulated NetBIOS</td>
<td>TCP/IP, IPX/SPX, Appletalk, TCP/IP encapsulated IPX, IPX encapsulated NetBIOS</td>
<td>TCP/IP, IPX/SPX</td>
</tr>
<tr>
<td>Routing supported</td>
<td>TCP/IP, IPX/SPX</td>
<td>TCP/IP, IPX/SPX, AppleTalk</td>
<td>TCP/IP, IPX/SPX</td>
</tr>
<tr>
<td>Remote access services</td>
<td>Windows RAS</td>
<td>Novell BorderManager</td>
<td>Optional</td>
</tr>
<tr>
<td>E-Mail gateways</td>
<td>Mail server optional</td>
<td>MHS included</td>
<td>SendMail</td>
</tr>
<tr>
<td>Clients able to access remote resources</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td><strong>Management &amp; Administration</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Can act as SNMP agent for enterprise mgmt system</td>
<td>yes</td>
<td>Optional</td>
<td>yes</td>
</tr>
<tr>
<td>Can set performance thresholds &amp; alerts</td>
<td>yes</td>
<td>yes with ManageWise (optional)</td>
<td>yes</td>
</tr>
<tr>
<td>Central mgmt of multiple servers</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Audit trails &amp; event logs</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
</tbody>
</table>

*Figure 9-10*  Server Network Operating System Technology Analysis Grid
given as an example but is not meant to be either authoritative or all-inclusive. The technology analysis grid is divided into the following major categories:

- Hardware/platform characteristics
- Installation and configuration
- Networking and connectivity
- Management and administration

**SERVER NETWORK OPERATING SYSTEM SELECTION**

New versions of server operating systems are released on an annual basis, if not more frequently. When in the market for a server operating system, it is important to consider all currently available products. You will want to consider current technology investments, cost of deploying the new technology, business objectives, required applications, as well as the stability and strategic product development direction of the operating system vendor. After completing this analysis, the best solution should be apparent.

**REMOTE ACCESS**

As the computing power available in portable formats has increased and the Internet has matured, the number of portable computers has increased. Combined with the evolution of n-tier client/server solutions, the need for these portable computers to gain remote access to corporate resources at off-site locations has also increased.

One of the most important things to understand about LAN remote access is the relatively limited bandwidth of the wide area network links that individuals will use to connect to corporate information resources. Although the goal of LAN remote access may be to offer transparent remote LAN connectivity, decreases in bandwidth by a factor of 100 on WAN links as compared to LAN links cannot be ignored.

**BUSINESS ISSUES OF REMOTE ACCESS**

As information has come to be seen as a corporate asset to be leveraged to competitive advantage, the delivery of that information to users working at remote locations has become a key internetworking challenge. Corporate downsizing has not only increased remaining employees’ responsibilities, but pushed those responsibilities ever closer to the corporation’s customers. As a result, the voice mail message, "I’ll be virtual all day today," is becoming more and more common. The business-oriented motivations for remote access to local LAN resources fall into about three general categories.

The first category of remote LAN access is often referred to as telecommuting, or more simply, working from home with all the information resources of the office LAN at one’s fingertips. This category of connectivity and computing is often referred to as SOHO, or Small Office Home Office.

Studies have indicated some of the ways in which telecommuting can increase overall worker productivity:

- Better, quicker, more effective customer service
- Increased on-time project completion and quicker product development
• Increased job satisfaction among highly mobile employees, which can lead to both greater productivity and employee retention
• Decreased worker turnover, which leads to decreased training and recruiting budgets
• Increased sales

A variation of telecommuting, mobile computing, addresses the need for field representatives to be able to access corporate information resources in order to offer superior customer service while working on the road. These field reps may or may not have a corporate office PC into which to dial.

Although some of the positive results of enabling remote access to corporate data for mobile workers are similar to those of telecommuters, the increased customer focus of the mobile worker is evident in the following benefits:

• Faster responses to customer inquiries
• Improved communications with co-workers and support staff at corporate offices
• Better, more effective customer support
• Increased personal productivity by the mobile workers such as being able to complete more sales calls
• Increased ability to be “on the road” in front of customers
• Allowing service personnel to operate more efficiently

The third major usage of remote computing is for technical support. Organizations must be able to dial in to client systems with the ability to appear as a local workstation, or take control of those workstations, in order to diagnose and correct problems remotely. Being able to diagnose and solve problems remotely can have significant impacts:

• Quicker response to customer problems
• Increased ability to avoid having to send service personnel for on-site visits
• More efficient use of subject matter experts and service personnel
• Increased ability to avoid re-visits to customer sites due to a lack of proper parts
• Greater customer satisfaction

**THE HIDDEN COSTS OF TELECOMMUTING**

In order to fully understand the total costs involved in supporting telecommuters, it is first essential to understand which employees are doing the telecommuting. Telecommuting employees generally fall into either one of the following categories:

• Full-time, day shift, at-home workers
• After-hours workers who have a corporate office but choose to extend the workday by working remotely from home during evenings and weekends
Most studies indicate that more than 75 percent of telecommuters are of the occasional, after-hours variety. However, corporate costs to set up and support these occasional users are nearly equal to the costs for setting up and supporting full-time at-home users—more than $4,000 per year. Among the hidden costs to be considered when evaluating the cost/benefit of telecommuting are the following:

- Workers might not be within local calling area of corporate resources, thereby incurring long-distance charges.
- The telephone company might have to add wiring from street to home or within home to support additional phone lines.
- If existing phone lines are used, personnel time is used to sort personal calls from business calls.
- In order to provide sufficient bandwidth, alternative access technologies such as ISDN, DSL, or cable modems are often installed, if available, because some applications, especially those not optimized for remote access, run very slowly over dial-up lines, leading to decreased productivity.

Architectural Issues of Remote Access

There are four steps to designing remote access capability for a network:

- Needs analysis
- Logical topology choice
- Physical topology choice
- Current technology review and implementation

Needs Analysis As dictated by the top-down model, before designing network topologies and choosing technology, it is essential to first determine what is to be accomplished in terms of LAN-based applications and use of other LAN-attached resources. Among the most likely possibilities for the information-sharing needs of remote users are the following:

- Exchanging e-mail
- Uploading and downloading files
- Running interactive application programs remotely
- Utilizing LAN-attached resources
- Attend virtual meetings via Internet telephony and conferencing

The purpose in examining information-sharing needs in this manner is to validate the need for the remote PC user to establish a connection to the local LAN that offers all of the capabilities of locally attached computers.

In other words, if the ability to upload and download files is the extent of the remote PC user’s information sharing needs, then file transfer software, often
Remote Access

included in asynchronous communications software packages, would suffice at a very reasonable cost. A network-based bulletin-board service (BBS) package is another way in which remote users can easily share information. Likewise, if e-mail exchange is the total information sharing requirement, then e-mail gateway software loaded on the LAN would meet that requirement.

However, in order to run LAN-based interactive application programs or utilize LAN-attached resources such as high-speed printers, CD-ROMs, mainframe connections, or FAX servers, a full-powered remote connection to the local LAN must be established. From the remote user’s standpoint, this connection must offer transparency: The remote PC should behave as if it were connected locally to the LAN. From the LAN’s perspective, the remote user’s PC should virtually behave as if it were locally attached.

**Logical Topology Choice: Remote Node vs. Remote Control**  
In terms of logical topology choices, two different logical methods for connection of remote PCs to LANs are possible. Each method has advantages, disadvantages, and proper usage situations. The two major remote PC operation mode possibilities are remote node and remote control.

The term **remote access** is most often used to generally describe the process of linking remote PCs to local LANs without implying the particular functionality of that link (remote node versus remote control). Unfortunately, the term **remote access** is also sometimes more specifically used as a synonym for remote node. Figure 9-11 outlines some of the details, features, and requirements of these two remote PC modes of operation, while Figure 9-12 highlights the differences between remote node and remote-control installations.

**Remote node** or remote client computing implies that the remote client PC should be able to operate as if it were locally attached to network resources. In other words, the geographic separation between the remote client and the local LAN resources should be transparent. In practice, the comparative bandwidth of a typical remote access link (ranging from 34-42 Kbps for a dial-up link to 512 Kbps for a DSL connection) compared with the Mbps bandwidth of the LAN is anything

<table>
<thead>
<tr>
<th>Functional Characteristic</th>
<th>Remote Node</th>
<th>Remote Control</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Also Called</strong></td>
<td>Remote client</td>
<td>Modem remote control</td>
</tr>
<tr>
<td></td>
<td>Remote LAN node</td>
<td></td>
</tr>
<tr>
<td><strong>Redirector hardware/ software required?</strong></td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td><strong>Traffic characteristics</strong></td>
<td>All client/server traffic</td>
<td>Keystrokes and screen images</td>
</tr>
<tr>
<td><strong>Application Processing</strong></td>
<td>On the remote PC</td>
<td>On the LAN-attached local PC</td>
</tr>
<tr>
<td><strong>Relative Speed</strong></td>
<td>Slower</td>
<td>Faster</td>
</tr>
<tr>
<td><strong>Logical role of WAN link</strong></td>
<td>Extends connection to an NIC</td>
<td>Extends keyboard and monitor cables</td>
</tr>
<tr>
<td><strong>Best Use</strong></td>
<td>With specially written remote client applications that have been optimized for execution over limited bandwidth WAN links</td>
<td>DOS applications; graphics on Windows apps; can make response time unacceptable</td>
</tr>
</tbody>
</table>

*Figure 9-11*  
Remote Node vs. Remote Control Functional Characteristics
but transparent. Where a NIC would normally plug directly into an expansion slot in a computer, a remote node connection merely extends that link via a relatively low speed connection. In remote node mode, client applications run on the remote client rather than a local LAN-attached client.

Client/server applications which require large transfers of data between client and server will not run well in remote node mode. Most successful remote node applications are rewritten to minimize large data transfers. For example, modified remote node e-mail client software allows just the headers of received messages, which include sender, subject, and date/time, to be transferred from the local e-mail server to the remote client. The remote client selects which e-mail messages should have the actual e-mail message body and attachments transferred. Local e-mail client software, which assumes plenty of LAN bandwidth, does not bother with such bandwidth conserving modifications. Other client/server applications must be similarly modified if they are to execute acceptably in remote node mode.

Although transparent interoperability was discussed as one of the goals of remote access, that does not necessarily mean that a worker’s mobile computer programs must be identical to those running on one’s desktop at the price of terrible performance. One of the most commonly overlooked aspects in deploying remote access solutions is the need to customize applications for optimal performance in a remote access environment.

Remote node mode requires a full client network operating system protocol stack to be installed on the remote client. In addition, wide area network communication software must be incorporated with the remote client NOS protocol stack. Remote node software often also includes optional support of remote control functionality.

Figure 9-12  Remote Node vs. Remote Control Installations
Remote control differs from remote node mode both in the technology involved and the degree to which existing LAN applications must be modified. In remote control mode, the remote PC is merely supplying input and output devices for the local client that interacts as normal with the server and other LAN resources. Client applications still run on the local client that is able to communicate with the local server at native LAN speeds, thereby precluding the need to rewrite client applications for remote client optimization.

Remote control mode requires only remote control software to be installed at the remote PC rather than a full NOS client protocol stack compatible with the NOS installed at the local LAN. The purpose of the remote control software is only to extend the input/output capabilities of the local client out to the keyboard and monitor attached to the remote PC. The host version of the same remote control package must be installed at the host or local PC. There are no interoperability standards for remote control software.

One of the most significant difficulties with remote control software is confusion by end users as to logical disk assignments. Recalling that the remote PC only supplies the keyboard and monitor functionality, remote users fail to realize that a C: prompt refers to the C: drive on the local LAN-attached PC and not the C: drive of the remote PC that they are sitting in front of. This can be particularly confusing with file transfer applications.

Protocols and Compatibility

At least some of the shortcomings of both remote node and remote control modes are caused by the underlying transport protocols responsible for delivering data across the WAN link. In the case of remote control, the fact that proprietary protocols are used between the guest and host remote control software is the reason that remote control software from various vendors is not interoperable. In the case of remote node, redirector software in the protocol stack must take LAN based messages and convert them into proper format for transmission over WAN links.

Figure 9-13 illustrates the protocol related issues of typical remote control and remote node links as well as TCP/IP based links for a dial-up environment. If a different access technology is used the required client to server communication technology would change accordingly (Cable Modem/CMTS, DSL Modem/DSLAM, etc.) per chapter 3.

Virtual Private Networks

In order to provide virtual private networking capabilities using the Internet as an enterprise network backbone, specialized tunneling protocols needed to be developed that could establish private, secure channels between connected systems. By using these tunneling protocols, a virtual private network (VPN) can be built between the remote workstation and the organization’s private network. A VPN creates an encrypted tunnel across a public network (typically the Internet) and passes the data destined for the remote location across the tunnel. The remote workstation gets a local IP address and appears to all computers on the local network as if it were local.

The two most commonly implemented VPN technologies are PPTP and L2TP/IPSec. L2TP/IPsec is largely supported by the firewall vendor community and is intended to provide interoperability between VPN firewalls from different vendors. PPTP is Microsoft’s tunneling protocol that is specific to Windows Servers and remote access servers. It has the backing of several remote access server vendors. Figure 9-14 illustrates the use of tunneling protocols to build virtual private networks across the Internet.
• **L2TP/IPSec.** Layer two tunneling protocol (L2TP) is an IETF standard based on Cisco Systems layer-two forwarding (L2F). L2TP is typically uses IPSec (secure IP) as an underlying encrypting protocol to secure the connection between the remote workstation and the VPN server. Once an encrypted IPSec connection is established across the public network, L2TP is used to create the actual VPN redirection and encapsulation.

• **PPTP—Point-to-point tunneling protocol.** PPTP is essentially just a tunneling protocol that allows managers to choose whatever encryption or authentication technology they wish to hang off either end of the established tunnel.
PPTP supports multiple network protocols including IPX, NetBEUI, and IP. PPTP is primarily concerned with secure remote access in that PPP enabled clients would be able to dial in to a corporate network via the Internet.

Remote Access Security Although security from an enterprise-wide perspective will be dealt with in chapter 12, security issues specifically related to remote access of corporate information resources are briefly summarized here. Security related procedures can be logically grouped into the following categories:

- **Password assignment and management.** Change passwords frequently, even considering single-use passwords. Passwords should not be actual words found in a dictionary, but should ideally be a random or meaningless combination of letters and numbers.

- **Intrusion responses.** User accounts should be locked after a pre-set number of unsuccessful logins. These accounts should only be able to be unlocked by a system administrator.

- **Logical/Physical Partitioning of Data.** Separate public, private, and confidential data onto separate physical servers to avoid users with minimum security clearances gaining unauthorized access to sensitive or confidential data.

- **Encryption.** Although it is important for any sensitive or proprietary corporate data to be encrypted, it is especially important that passwords be encrypted to avoid interception and unauthorized re-use.

- **Dial-back systems.** After remote users enter proper UserID and passwords, these systems terminate the call and dial the authorized user back at pre-programmed phone numbers.
Remote Client Software Authentication Protocols. Remote client protocol stacks often include software-based authentication protocols such as PAP (Password Authentication Protocol) or CHAP (Challenge Handshake Authentication Protocol).

Remote Client Authentication Devices. Although exact implementation details may vary from one vendor to the next, all token authentication systems include server components linked to the communications server, and client components which are used with the remote access clients. Physically, the token authentication device employed at the remote client location may be a hand-held device resembling a calculator or just a small LCD screen capable of displaying six digits, a floppy disk, or it may be an in-line device linked to either the remote client’s serial or parallel port. Token authentication devices are explained further in chapter 12.

Physical Topology: Alternative Access Points As Figure 9-15 illustrates, there are two basic ways in which a remote PC user can gain access to the local LAN resources.

- Serial port of a LAN-attached PC. Perhaps the simplest physical topology or remote access arrangement is to establish a communications link to a user PC located in the corporate office. However, many field representative or mobile computing users no longer have permanent offices and workstations at a corporate building and must depend on remote access to shared computing resources.

- Communications-server. As an alternative to having a dedicated PC at the corporate office for each remote user to dial into, remote users could attach to a dedicated multi-user server, known as a remote access server or communications server through one or more modems or via a VPN connection across the Internet. Depending on the software loaded on the communications server, it may deliver remote node functionality, remote control functionality, or both. As telecommuting and demand for internet access have increased, remote access servers have become the dominant means for accessing networks remotely.

The physical topology using the communications server (Figure 9-15, Illustration 2) actually depicts two different possible remote LAN connections. Most communications servers answer the modem, validate the UserID and password, and log the remote user onto the network. Some communications servers go beyond this to allow a remote user to access and/or remotely control a particular networked workstation. This scenario offers the same access capabilities as if the networked workstation had its own modem and software, but also offers the centralized management, security, and possible financial advantage of a network attached communications server.

The access arrangements illustrated are examples of possible physical topologies and do not imply a given logical topology such as remote node, remote control, or both. It is important to understand that the actual implementation of each of these LAN access arrangements may require additional hardware and/or software. They may also be limited in their ability to utilize all LAN attached resources.

Network Topology: Alternative Network Access Services While Figure 9-15 illustrated alternative access points within an enterprise network, numerous network access
service alternatives exist that a remote access client can employ to reach an enterprise network’s access point. Among these alternatives are the following, many of which were introduced in chapter 3.

- **Public Switched Telephone Network (PTSN).** Switched analog service, requires a modem. The highest current modem standard is V.90, theoretically 56 Kbps, closer to 33 Kbps in reality.
- **Integrated Services Digital Network (ISDN).** Switched digital service, requires an ISDN “modem,” 64 Kbps per B channel; it may be able to combine 2 B channels for 128 Kbps.
- **Digital Subscriber Line (xDSL).** This fixed point-to-point digital service rides over PSTN circuit. Requires DSL “modem.” Bandwidth ranges from 64 Kbps to 1.5 Mbps; may be symmetrical or asymmetrical.
- **Cable Modems.** This Internet access technique provides high bandwidth at a low price point. However many providers block the ability to use VPN technologies over the link, thereby reducing their usefulness for remote access solutions.
- **Virtual Private Network (VPN).** Builds secure communication channels through the Internet to connect remote corporate sites with the regional or headquarters sites. Requires VPN hardware and software and access to an Internet service provider.

Figure 9-16 illustrates alternative Network Access Services that might be used to construct a remote access network topology.
Remote Access Technology

There are two key components to a remote access solution: software and hardware. Remote access software consists of communications servers and remote access servers. Required hardware includes NICs and any other hardware required to provide a connection from the server to the client. The following sections provide a closer look at these required components.

Communications Servers and Remote Access Servers  As is often the case in the wonderful but confusing world of data communications, communications servers are also known by many other names. In some cases these names may imply, but don’t guarantee, variations in configuration, operation, or application. Among these varied labels for the communications servers include access servers, remote access servers, and remote node servers.

A communications server offers both management advantages as well as financial payback when large numbers of users wish to gain remote access to and from a LAN. Besides the cost savings of a reduced number of computers, modems, phone lines and other access hardware, there is a significant gain in control of remote access to the LAN and its attached resources.

Multiple remote users can access a communications server simultaneously. Exactly how many users can gain simultaneous access will vary with the type of connection being used (dial-up modem, Internet, or VPN). Most communications servers service at least four simultaneous users—as many as 1,000 are possible.

Remote Control Software  Remote control software, especially designed to allow remote PCs to “take-over” control of local PCs, should not be confused with the
asynchronous communications software used for dial-up connections to asynchronous hosts via modems. Modem operation, file transfer, scripting languages and terminal emulation are the primary features of asynchronous communications software.

Taking over remote control of the local PC is generally only available via remote control software. Remote control software allows the keyboard of the remote PC to control the actions of the local PC with screen output being reflected on the remote PC’s screen. The terms remote and local are often replaced by guest (remote) and host (local).

Operating remote control software requires installation of software programs on both the guest and host PCs. Various remote control software packages do not interoperate. The same brand of remote control software must be installed on both guest and host PCs. Both the guest and host pieces of the remote control software may or may not be included in the software package price. Remote control software must have modem operation, file transfer, scripting language and terminal emulation capabilities similar to those of asynchronous communications software. However, in addition, remote control software should perform the following functions:

- Avoid lockups of host PCs
- Allow the guest PC to disable the keyboard and monitor of the host PC
- Add security precautions to prevent unauthorized access.
- Include virus detection software

Additionally, Windows-based applications pose a substantial challenge for remote control software. The busy screens of this graphical user interface can really bog down communications links, especially if large screen resolutions or color depths are specified. Some remote control software solutions have included the capability to reduce the number of bits of color depth to reduce the amount of data that must be sent across the communication link. Figure 9-17 summarizes the important features of remote control software, as well as their potential implications.

The Remote Control Software loaded onto a communications server for use by multiple simultaneous users is not the same as the remote control software loaded onto single remote (guest) and local (host) PCs. Communications Servers’ remote control software has the ability to handle multiple users, and in some cases, multiple protocols.

Remote Node Software Traditionally remote node client and server software were supplied by the vendor of the network operating system on the server to be remotely accessed. Windows RAS (Remote Access Service) and NetWare BorderManager are two examples of NOS-specific remote node server software. It is important to note that these are software-only solutions, installed on industry standard, Intel application servers, as opposed to the proprietary hardware of specialized remote access or communications servers.

Some of the important functional characteristics of remote node server software other than operating system/network operating system compatibility are listed in Figure 9-18.
### Feature Category: Protocol Compatibility
- **Network Operating System Protocols**
  - Which network operating system protocols are supported? (IP, IPX, NetBIOS)

### Feature Category: LAN Compatibility
- **LAN versions**
  - Are specific multi-user LAN server versions available or required?
- **Host/guest**
  - Are both host & guest (local & remote) versions included?
- **Operating system**
  - Some remote control packages require the same operating system at host and guest PCs while others do not.

### Feature Category: Operational Capabilities
- **Printing**
  - Can remote PC print on local or network attached printers?
- **File transfer**
  - Which file transfer protocols are supported?
  - Delta file transfer allows only changes to files to be transferred.
  - Automated file and directory synchronization is important to mobile workers who also have desktop computers at home or at the office.
  - Allows repetitive call set-ups and connections to be automated.
- **Scripting language**
  - Can the software dynamically reduce the color depth to save bandwidth?
- **Color depth/ resolution**
  - Some packages allow more than one connection or more than one session per connection, for example, simultaneous file transfer and remote control.

### Feature Category: Security
- **Password access**
  - This should be the minimum required security for remote login.
- **Password encryption**
  - Since passwords must be transmitted over WAN links it would be more secure if they were encrypted.
- **Keyboard disabling**
  - Since the local PC is active but controlled remotely, it is important that the local keyboard be disabled to prevent unauthorized access.
  - Similar to rationale for keyboard disabling, since output is being transmitted to the remote PC it is important to blank the local monitor so that processing cannot be viewed without authorization.
- **Monitor blanking**
  - If a dial-in solution is being implemented call back can add security. Although not hacker-proof, the server hangs up on dial in, and calls back at pre-programmed or entered phone number.
- **Call-back system**
  - Are remote users able to be restricted to certain servers, directories, files, or drives? Can the same user be given different restrictions when logging in locally or remotely?
- **Access restriction**
  - Can system managers or enterprise network management systems be notified when remote access or password failures have occurred?
  - Can the remote PC (guest) reboot the local host if it becomes locked up?
  - Are users locked out after a given number of failed login attempts?
- **Remote access notification**
  - This feature is especially important given file transfer capabilities from remote users.
- **Remote host reboot**
  - Can remote users be restricted to read-only access?
- **Limited logon attempts**
  - In order to save on long distance charges, can users be logged off (and calls dropped) after a set length of time?

*Figure 9-17* Remote Control Software Technology Analysis
Mobile-Aware Operating Systems The mobile computer user requires flexible computing functionality in order to easily support at least three possible distinct computing scenarios:

- Stand-alone computing on the laptop or notebook computer
- Remote node or remote control computing to corporate headquarters
- Synchronization of files and directories with desktop workstations at home or in the corporate office

Operating systems that are able to easily adapt to these different computing modes with a variety of included supporting accessory programs and utilities are sometimes referred to as mobile-aware operating systems. Most modern client operating systems offer some sort of native mobile-awareness. Among the key functions offered by such mobile-aware operating systems are the following:

- **Auto-detection of multiple configurations.** If external monitors or full-size keyboards are used when at home or in the corporate office, the operating system should automatically detect these and load the proper device drivers.
• **Built-in multi-protocol remote node client.** Remote node software should be included which can automatically and transparently connect to the network via any available connection including: wired network connections, wireless LAN connections, and dial-up modem connections.

• **File transfer and file/directory synchronizations.** Once physical connections are in place, software utilities should be able to synchronize files and directories between either the laptop and the desktop or the laptop and the corporate server.

• **Deferred printing.** This feature allows printed files to be spooled to the laptop disk drive and saved until the mobile user is next connected to corporate printing resources. At that point, instead of having to remember all of the individual files requiring printing, the deferred printing utility is able to automatically print all of the spooled files.

• **Power management.** Since most mobile computing users depend on battery-powered computers, anything that the operating system can do to extend battery life would be very beneficial. The demand for higher-resolution screens has meant increased power consumption in many cases. Power management features offered by operating systems have been standardized as the Advanced Power Management (APM) and Advanced Configuration and Power Interface (ACPI) specification.

**SUMMARY**

Network operating systems have traditionally provided shared file and print services among networked clients. With the increase in client/server architectures and the associated increase in distributed applications, network operating systems are now also providing application services, directory services and messaging and communications services in support of these distributed applications.

Client network operating systems functionality can be categorized into operating systems capabilities, peer-to-peer networking capabilities, and client networking capabilities. Client networking capabilities are largely measured by the number of different server network operating systems with which the client can transparently interoperate. Remote access capability is also important.

Server network operating systems are now primarily concerned with high performance application services for back-end application programs. Enterprise-wide directory services must also be provided. The two major approaches to enterprise directory services are global directory services and domain directory services.

In order to communicate with numerous client platforms, server network operating systems must support a variety of different network clients as well as a variety of different network transport protocols. Multi-protocol routing and remote access services are also essential to deliver transparent interoperability to the greatest number of client platforms. In the multiple server environments of the enterprise network, monitoring, management and administration tools play a critical role.

Remote access to LANs has taken on increased importance in response to major changes in business conditions. As indicated by the top-down model, network functionality must respond to changing business conditions. Expectations of LAN remote access are significant. Remote users expect the same level of data accessibility, application services, and performances on the road as they receive at the office. Delivering this equivalent functionality is the challenge faced by networking professionals today. The major obstacle to this objective is bandwidth, availability, and quality of the wide area network services which are expected to deliver remote connectivity to mobile users.
There are two basic logical topologies for remote access. Remote control allows a remote PC to take over or control a local PC. Processing occurs on the local PC and only keyboard strokes and screen images are transported over the WAN link. Remote node allows the remote PC to act as a full-fledged LAN client to the local LAN server. In this case, full client/server traffic travels over the WAN link as the application executes on the remote client PC. One of these logical topologies is not preferable in all cases. Each situation must be analyzed on an individual basis.

Physical topologies include accessing a local LAN-attached PC directly via modem, accessing a shared communications server via a dedicated wide area network connection, or connecting to the LAN via a virtual private network across a public access network such as the Internet.

**KEY TERMS**

16-bit sub-system  
access control list (ACL)  
access server  
application program interface  
application services  
authentication  
authentication credentials  
authorization  
auto-detection & configuration  
Challenge/Handshake Authentication Protocol (CHAP)  
client network operating systems  
client/server network operating communications server  
delta file transfer  
dial-in server  
dial-up server  
directory services  
directory synchronization software  
domain services  
domains  
encryption  
file synchronization software  
lightweight directory access protocol (LDAP)  
NetWare directory services (NDS)  
network objects  
password authentication protocol (PAP)  
peer-to-peer network operating systems  
performance monitoring  
plug-n-play  
PnP  
PnP BIOS  
RAS  
remote access  
remote control  
remote control software  
remote node  
remote node client software  
remote node server  
remote node server software  
remote node servers  
remote node software  
screen caching  
server network operating  
small office home office (SOHO)  
telecommuting  
universal client  
virtual machines  

**REVIEW QUESTIONS**

1. Differentiate between peer-to-peer network operating systems and client/server network operating systems.
2. How does the combination of today’s client and server network operating systems differ from a traditional client/server network operating system implementation?
3. What is a universal client?
4. Why is a universal client important to enterprise computing?
5. What new demands for services are being put on today’s server network operating systems?
6. Describe the importance of the following service categories in more detail: directory services, application services, integration/migration services.
7. Describe the major categories of functionality of client network operating systems.
8. What is the objective of PnP standards?
9. Describe the components required to deliver a PnP solution, and the relationship of the described components.
10. Describe the three elements of networking functionality belonging to client network operating systems paying particular attention to the relationship between the elements.
11. Why is it important for a client network operating system to be able to support more than one network transport protocol?
12. Describe the importance of laptop synchronization as a client network operating system feature.
13. Describe the major differences between global directory services and domain services in terms of architecture and functionality.
14. What is LDAP?
15. What is relationship between file systems, APIs and application services?
16. What are the two basic parts of an authentication credential set?
17. What is the difference between authentication and authorization?
18. What is the purpose of an ACL?
19. Why might it be important for a network operating system to support more than one file system?
20. What is the role of NCP, SMB, and NFS redirectors in offering application services?
21. What are some important functional characteristics of server network operating systems related to installation and configuration?
22. What are some important functional characteristics of server network operating systems related to integration and migration?
23. Name and describe the issues surrounding at least 4 areas of NOS functionality which must be addressed when designing interoperability solutions.
24. What are some of the key business trends which have led to an increased interest in LAN remote access?
25. What is the importance of needs analysis to LAN remote access design?
26. Differentiate between remote node and remote control in terms of functionality and network impact.
27. What is the major limitation in terms of delivering transparent access to remote LAN users?
28. Describe how it is possible to run remote control software via a remote node connection. What are the advantages of such a setup?
29. What are some of the security issues unique to remote access situations?
30. What is the relationship between the guest and host remote control software?
31. Differentiate between remote control and remote node software in terms of transport protocols and client protocol stacks.
32. What are some of the unique functional requirements of remote node server software?

---

Case Study: For a business case study and questions that relate to network services, go to www.wiley.com/college/goldman.