

# Part 1

## Building a Foundation

### Topics Covered:

- Exchange 2000 Architecture
- Picking the Right Edition
- Storage Considerations
- Synchronization and Standardization
- Server Hardware Considerations
- User Education
- Active Directory and DNS
- Basics of Active Directory Operation
- Preparing the Active Directory Forest and Each Domain
- Exchange 2000 and Active Directory Groups
- Customizing Active Directory
- Importing and Modifying Active Directory Data
- Preparing for a Migration/Upgrade to Exchange 2000
- Understanding the Active Directory Connector
- The Site Replication Service and the Recipient Update Service



# 1

## Getting Started with Exchange 2000

**A** successful Exchange 2000 deployment hinges on many elements; these include a strong dependency on Windows 2000 Active Directory (AD), Windows 2000 Internet Information Server (IIS), a properly configured DNS (Domain Name Service) infrastructure, sufficient and reliable hardware, and good operational practices. Your Exchange 2000 installation will have serious problems unless you have a good understanding of not only Exchange 2000, but also Windows 2000, AD, and DNS. Exchange 2000's destiny is much more intertwined with Windows 2000 than earlier versions of Exchange. A basic understanding of the Exchange 2000 architecture and deploying Exchange 2000 on the proper hardware platform will also be crucial to your success.

One of the most important parts of deploying any Exchange system is making design decisions that relate to supporting your organization. This includes choosing the right edition of Exchange 2000 Server, deciding how to best store your data, maintaining time synchronization, setting reasonable standards (Active Directory, Exchange performance, user space allocation, etc.), and picking the right hardware. Placing your Exchange 2000 system on appropriately sized and configured hardware will also help to keep you happy and safe from end-user lynch mobs.

Finally, providing your user community with good documentation, notification, and training will help to minimize your administration woes. Most experienced Exchange administrators will tell you that educating their users, keeping them informed, and managing their expectations are some of the most powerful tools in their operations arsenal.

Yet perhaps first and foremost, essential tools to have in your bag of tricks are solid operations practices that will help reduce the likelihood of downtime and improve the recoverability from disasters—and help keep you sane. One particularly wise Exchange guru once said his secret to Exchange success was the following:

- Perform daily backups of Exchange.
- Check the event logs.
- Make sure the server does not run out of disk space.
- Check the queues.
- Then leave Exchange alone.

So, is that all there is to say about Exchange administration? If so, why have volumes of information been written about it, and why am I writing more? The answer is simple: We all benefit from shared experiences. Combine that with the fact that software documentation and training do not always make matters crystal clear, and you have good reasons for a book about skillfully maintaining Exchange.

#### **The Makings of a Good Exchange Administrator**

I have tended to a number Exchange “disasters” where the clients were running either Exchange 5.5 or Exchange 2000. These were situations in which I was called in to fix a pretty serious problem. I classify these as disasters because in each case the user community was without e-mail services for more than half of a business day. In one case, the user community was without e-mail for more than a week before I was called. One of the strengths I look for in system administrators is the ability to know when they are in over their heads and when to call for help. This includes not being afraid to call Microsoft Product Support Services.

With a few exceptions, the aforementioned disasters were either caused or compounded by administrators who were not prepared for the disaster, did not know what they were doing, or did not call for help when they should have. The administrators did not have a clear understanding of Exchange, Active Directory, and the steps to successfully manage an Exchange system, nor had they documented or practiced disaster recovery beforehand.

Disaster prevention involves two major steps. The first is recognizing that you cannot solve every problem in the world (and not being afraid to admit it). The second step—and the one you are taking now, by reading this book—is to do everything you can to improve your knowledge of Exchange 2000 (and Windows 2000).

## Exchange 2000 Architecture Basics

If you lived the life of an Exchange 5.5 administrator, you should have a strong understanding of the primary differences between Exchange 5.5 and Exchange 2000. Any time a systems administrator or network engineer is troubleshooting a problem, knowledge of the system's architecture as well as of the system dependencies is essential.

### Exchange 2000 for Exchange 5.5 Administrators

The differences between Exchange 5.5 and Exchange 2000 are staggering considering that the two products are merely one generation apart. For one, Exchange 5.5 is a complete messaging system. It includes not only a message store, but also a directory service and components to deliver messages to other systems. Also, Exchange 5.5 relies on Windows NT as an operating system platform and for authentication.

Exchange 2000 still relies on Windows 2000 as an operating system and for authentication, but essentially Exchange 2000 is a message storage engine with interfaces to AD and IIS. Directory services and message transport is no longer the concern of Exchange 2000; they have been offloaded to Windows 2000. There are many other differences between Exchange 5.5 and Exchange 2000, such as more storage scalability, message routing, and an improved Outlook Web Access (OWA) interface.

### Hey, Who Took My Directory?

The most striking change between Exchange 5.5 and 2000 Server is that Exchange 2000 does not have its own directory service. Rather, Exchange 2000 relies entirely on Windows 2000 Active Directory. Some of the information that is now stored in AD includes:

- User-specific information, such as telephone number, title, department, e-mail addresses, Instant Messaging home server, and mailbox home server
- Access Control Lists (ACLs) for accessing mailboxes, public folders, and administrative rights
- Server configuration, including what mailbox and public folder stores are on each server, SMTP configuration, connectors, routing group configuration, and administrative group configuration
- Exchange 2000-specific classes of objects and attributes of those objects that are unique to Exchange

Also, Active Directory, not Exchange 2000, now controls all replication of user and configuration data. In my opinion, AD is the most important component of Exchange 2000, even though it is actually part of Windows 2000.

---

**NOTE** Active Directory is covered in more detail in Chapter 2, “Active Directory for Exchange 2000 Administrations.”

### Message Transport

The Exchange 5.5 message transfer agent (MTA) is responsible for delivery of all messages that are “leaving” the server, whether the message is going to a connector (such as the Internet Mail Service), another Exchange server, another Exchange site, or a foreign connector such as X.400. Further, the Exchange 5.5 MTA expands recipients of distribution lists.

In Exchange 2000, SMTP is now the message transport used for most inter-server message transport. The Exchange 2000 MTA is responsible for delivery of messages to Exchange 5.5 servers, Exchange 2000 routing groups connected via X.400, and foreign X.400 Connectors.

Unlike Exchange 5.5, all messages delivered by Exchange 2000 are routed through the Advanced Queuing Engine, a component of Internet Information Server. This includes local deliveries as well as messages destined for other Exchange servers (both 5.5 and 2000), distribution lists, foreign connectors, third-party gateways, and the Internet. The Advanced Queuing Engine is covered in more detail later in this chapter.

### Is It a Site, a Routing Group, or an Administrative Group?

An Exchange 5.5 site serves two primary purposes. First, the site is a boundary of high-speed, full-time connectivity. Second, the site is a boundary of administrative control.

Exchange 2000 introduces the concept of *administrative groups* and *routing groups*, which effectively separate administrative permission requirements and message routing requirements. The administrative group is a boundary of administrative control; the routing group is a boundary of reliable, full-time connectivity. In a native Exchange 2000 organization, administrative and routing groups can have different boundaries; a server can be in one administrative group but in a routing group that is contained in another administrative group. Grouping servers (for administrative purposes) based on bandwidth is no longer necessary.

### Waiter, There’s an STM File in My Data Directory!

All Exchange 5.5 message data is stored in either the PUB.EDB file or the PRIV.EDB file, depending whether the message was to be stored in the public or private information store, respectively. With the release of Exchange 2000, message data is now split between

two types of files: the EDB file and the STM file. Each public and private information store has an STM file and an EDB file.

- The EDB file is also known as the *rich-text store* or *MAPI store*; I have even seen it referred to as the *property store*. This file contains all messages sent by MAPI clients as well as folder content listings, indexes, and a subset of properties of each message stored in the STM file.
- The STM (streaming) file is also known as the *native content store*. Messages that are sent to Exchange by clients other than Outlook, such as from MIME (Multipurpose Internet Mail Extensions) or non-MIME clients, are stored in this file, as are messages inbound from the SMTP service. For performance reasons, only a subset of properties is converted to and stored in the MAPI store.

Storage of data between these two files is transparent to the client. When a client retrieves a message located in either file, the message content is converted to the appropriate format as necessary and passed on to the client.

### Where Are Link Monitor and Server Monitor?

One of my favorite (and often one of the most unreliable) Exchange 5.5 tools is the Exchange Administrator's Link Monitor. Running a close second is Server Monitor. These two tools let me ensure that messages are flowing to different parts of my organization and that services I have designated are running on my Exchange servers. However, historically these tools have been inaccurate due to slow WAN links, message delays, and incorrect administrative permissions.

Exchange 2000 has replaced Server Monitor and Link Monitor with the other tools found in the Exchange System Manager console in the Tools > Monitoring And Status folder. The Notifications and Status tools allow you to monitor the availability of connectors, the availability of servers, queue growth, free disk space, and other Windows 2000 services. Though I originally liked the old tools better, I have now found the new tools to be both reliable and flexible.

### Other Enhancements and Changes

Other enhancements and changes found in Exchange 2000 include:

- The Gateway Address Routing Table (GWART) has been replaced by the much more robust Link State Table, which includes information about whether a specific route is currently available.
- All administration is now performed through Microsoft Management Console (MMC) snap-ins. These can be run from any Windows 2000-based computer.
- OWA has been completely rewritten to use more efficient, modern Internet technologies such as HTTP/DAV (Distributed Authoring and Versioning), DHTML (Dynamic HyperText Markup Language), and XML (Extensible Markup Language).

## Exchange 2000 Dependencies

There are a number of Windows 2000 services that must be started before you can start the first Exchange service. There are still other services that Exchange 2000 requires to be installed prior to Exchange installation (IIS services including the web service, SMTP, and NNTP.) Table 1.1 lists the Windows 2000 services that must be started in order to start the principal services.

**Table 1.1** Exchange 2000 Dependencies

Exchange 2000 Service	Windows 2000 Services (Dependencies)
Microsoft Exchange System Attendant (mad.exe)	Event log NT LM Security Support Provider Remote Procedure Call (RPC) Remote Procedure Call (RPC) Locator Server Workstation
Microsoft Exchange Information Store (store.exe)	IIS Admin Service Microsoft Exchange System Attendant
Microsoft Exchange MTA Stacks (emsmta.exe)	IIS Admin Service Microsoft Exchange System Attendant
Microsoft Exchange IMAP4 (part of inetinfo.exe)	IIS Admin Service Microsoft Exchange Information Store
Microsoft Exchange POP3 (part of inetinfo.exe)	IIS Admin Service Microsoft Exchange Information Store
Simple Mail Transport Protocol (SMTP) (part of inetinfo.exe)	IIS Admin Service (The SMTP service is actually part of Windows 2000, but is enhanced during the installation of Exchange 2000.)
Network News Transport Protocol (NNTP) (part of inetinfo.exe)	IIS Admin Service (The NNTP service is actually part of Windows 2000, but is enhanced during the installation of Exchange 2000.)
Microsoft Exchange Event (events.exe)	Microsoft Exchange Information Store

**Table 1.1** Exchange 2000 Dependencies *(continued)*

Exchange 2000 Service	Windows 2000 Services (Dependencies)
Microsoft Exchange Routing Engine (part of inetinfo.exe)	IIS Admin Service
Microsoft Search (mssearch.exe)	NT LM Security Support Provider Remote Procedure Call (RPC)
Microsoft Exchange Key Management Service (kmservice.exe)	Microsoft Exchange Information Store
Microsoft Exchange Site Replication Service (srsmain.exe)	Event log NT LM Security Support Provider Remote Procedure Call (RPC) Remote Procedure Call (RPC) Locator Server Workstation
Microsoft Exchange Connectivity Controller (lscntrl.exe)	Event log Microsoft Exchange System Attendant
Microsoft Exchange Connector for Lotus cc:Mail (ccmc.exe)	Event log Microsoft Exchange Information Store
Microsoft Exchange Connector for Lotus Notes (dispatch.exe)	Event log Microsoft Exchange Connectivity Controller Microsoft Exchange Information Store
Microsoft Exchange Directory Synchronization (dxa.exe)	Microsoft Exchange MTA Stacks
Microsoft Exchange Router for Novell GroupWise (gwrouter.exe)	Event log
MS Mail Connector Interchange (mt.exe)	Event log Microsoft Exchange MTA Stacks
MS SchedulePlus Free-Busy Connector (msfbconn.exe)	Event log Microsoft Exchange Information Store

**Table 1.1** Exchange 2000 Dependencies (*continued*)

Exchange 2000 Service	Windows 2000 Services (Dependencies)
Microsoft Exchange Connector for Novell GroupWise (dispatch.exe)	Event log Microsoft Exchange Connectivity Controller Microsoft Exchange Information Store Microsoft Exchange Router for Novell GroupWise
Microsoft Active Directory Connector (adc.exe)	Event log NT LM Security Support Provider Remote Procedure Call (RPC) Remote Procedure Call (RPC) Locator Server Workstation

Other network services that must be available in order for Exchange 2000 to function properly include:

- Windows 2000 domain controller
- Windows 2000 Global Catalog server
- DNS server that will resolve service records for the Windows 2000 forest, MX (mail exchanger) records, and A (address or host) records

## An In-Depth Look at Exchange 2000 Architecture

Now that we have covered the basics, let's take a closer look at the Exchange 2000 architecture. This includes a more in-depth examination of Exchange 2000 message storage, message access with Exchange Installable File System (ExIFS) and IIS, and Advanced Queuing Engine, as well as connectors and the System Attendant service.

### Message Storage

A number of improvements were made in the database engine that ships with Exchange 2000; the current database engine is called ESE98 (Extensible Storage Engine). More information on the database engine and storage technology is in Chapter 4, "Maintenance and Management."

Exchange breaks down storage into either public or private information stores. All mailbox data is stored in a mailbox store (private information store) while all public folder data is stored in a public folder store, and these stores each have two separate components, an EDB file and an STM file.

**NOTE** A MAPI (Messaging Application Programming Interface) client is any client that sends and reads messages where the message properties are defined as MAPI properties. MAPI clients include the original Exchange client, Outlook 97/98/2000/ and Outlook 2002.

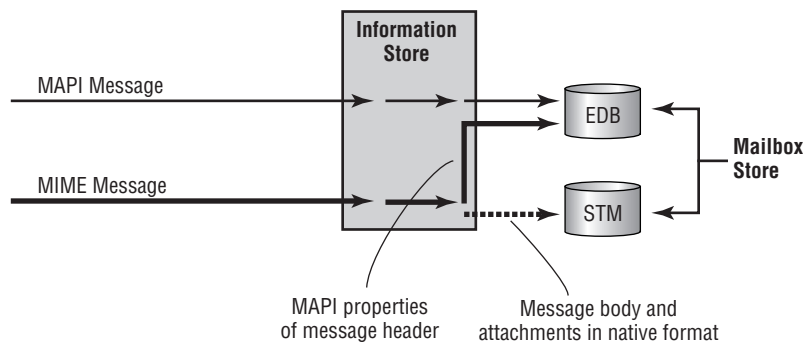
As mentioned earlier in this chapter, the EDB file is called the MAPI store. This is a rich, hierarchical property store; messages sent by MAPI clients are stored here. Thus all messages stored here have MAPI properties associated with them.

The STM file is not nearly as structured as the EDB file. Messages are not converted to MAPI messages when they arrive, but instead are stored in their native format (typically MIME). This includes messages sent by SMTP clients. However, the EDB file does contain a list of *all* messages stored in each folder, so certain MAPI properties for messages in the STM file are promoted to the EDB file. To improve performance, the STM file data is accessed through a kernel-mode device driver called ExIFS; a Windows Explorer extension that uses this device driver also allows the entire store to be accessible through the file system.

**NOTE** By now, you have probably seen the term “Web store” or “Web storage system” used (or overused) in technology media. The Web store is not actually a single database but a technology for providing access to data through HTTP/DAV or the ExIFS.

Figure 1.1 illustrates two examples of message storage, a MAPI message and a MIME message.

**Figure 1.1** Messages arriving in a mailbox store



The first message shown in Figure 1.1 is sent by a MAPI client such as Outlook 2000. The client designates most of the message's MAPI properties, and the client sends the message to the Exchange server; the message transport and the store may also set some of the message properties. The information store saves the entire message in the EDB file.

The second message is formatted by a client such as Outlook Express as a MIME message. The Internet Mail Service in Exchange 5.5 would have converted this message, but the Advanced Queuing Engine in Exchange 2000 simply passes it along to the information store in its native format. The information store determines that the message is in native format, then "promotes" certain properties of the message header (such as the To, From, Subject, and Date information) to MAPI properties, and finally stores this information in the EDB file along with a "pointer" that points to the message body and attachments in the STM file. Technically, there are actually three separate phases of property promotion: The initial properties are promoted when the message is sent to the server by the client, the second set when the messages is accessed, and finally if the content is changed by a MAPI client.

---

**NOTE** In a pure MAPI environment with no SMTP connectivity to the outside world, your STM files will hardly grow in size at all. In an environment with all POP3 and IMAP4 clients, the STM file will grow significantly while the EDB file will hardly increase in size.

### Content Conversion on Demand

The obvious question now is "What happens if a MAPI client reads a message that was sent to the Exchange server via SMTP and is formatted as a MIME message?" Simple. The information store retrieves the message into memory on the Exchange server and performs an "on-the-fly" conversion. The message is *not* converted in the STM file, merely in the copy in memory. The message is saved as a MAPI message only if a MAPI client modifies the message. If the message contains an attachment but the attachment is not modified, then it is not moved into the EDB data file.

The same holds true of a message that was sent by a MAPI client but is now being retrieved by a non-MAPI client such as Outlook Express. The information store converts the message on-the-fly to a MIME or non-MIME message and passes it on to the client.

So why all the conversion? Why not just store all messages in a common format? In Exchange 5.5, all inbound SMTP message content was converted to MDBEF message format by the information store's IMAIL process. If the message was retrieved by a POP3 or IMAP4 client, it was once again converted by the IMAIL process. If your environment is a pure environment of one type or another (all MAPI or all MIME clients), converting to another format would be too much overhead compared to keeping the content in its native format.

Microsoft's developers recognize the changing nature of the messaging world and that in the future we will have more mixed-client environments, which achieved better performance than if they simply stored the message in its native format and converted the message only when necessary. OWA and IMAP4 clients are becoming increasingly popular, and future versions of Outlook will more than likely provide the ability to access data using HTTP/DAV rather than MAPI. With a steady turn toward an emphasis on XML, HTTP/DAV, and other "Internet" clients, it makes sense to figure out how to keep data stored in its native format without content conversion. Further, the streaming store provides much higher performance access for message attachments. Messages stored in the EDB file are written in 4K page reads, whereas the STM file is accessed using kernel-level I/O in 64K streamed chunks. This is much more efficient.

### Storage Groups and Multiple Stores

In Exchange 5.5, you are limited to a single private information store and a single public information store. If you are running Exchange 2000 Enterprise Server, you can create up to 20 separate mailbox or public folder stores. Storage groups are used to organize these mailbox and public folder stores. Exchange 2000 Server allows for only a single mailbox store (maximum EDB size of 16GB), but up to four public folder stores.

**Storage Groups** Storage groups are the building blocks for multiple stores. Exchange 2000 Enterprise Server allows you to create up to four separate storage groups, each of which can contain up to five mailbox stores or public folder stores and has its own set of transaction logs. Circular logging can be turned on for some storage groups depending on the requirements of the data stored in the storage group.

---

**TIP** For optimal performance, each storage group's transaction log files should be placed on a separate physical hard disk. The transaction logs should not share this hard disk with any other application or data.

When the first database in a storage group is mounted, a new instance of the ESE database engine is started. All instances of ESE run as part of the `store.exe` process.

**Multiple Stores** What possible uses can there be for additional mailbox stores? Here is a list of possible advantages to using more than one mailbox store:

- Company executives or VIPs can be placed in a separate mailbox store to allow for quicker backup and restoration times.
- The overall size of any specific mailbox store can be reduced by splitting up the storage load between two stores.

- You can specify separately which stores need to be full-text indexed and which do not.
- Additional public folder stores can be used to store data that is accessed exclusively via OWA or the ExIFS driver.

However, be cautioned that if you choose to have more than one private mailbox store on a single server, there are a few things you should consider:

- Each additional store that you mount consumes at least another 10MB of RAM.
- Single-instance storage is preserved only within a single store. Recipients across multiple stores will cause multiple copies of a message to be created.
- Backup and recovery scenarios require more diligence and testing in this more complicated environment.

## Message Access

Exchange 2000 introduces an entirely new approach to accessing message storage. In addition to being able to access message and public folder data through MAPI, POP3, IMAP4, NNTP, and HTTP clients, you can now access Exchange data directly through the file system. (The POP3, IMAP4, NNTP, and HTTP processes are all part of IIS.)

### Exchange 2000 Installable File System (ExIFS)

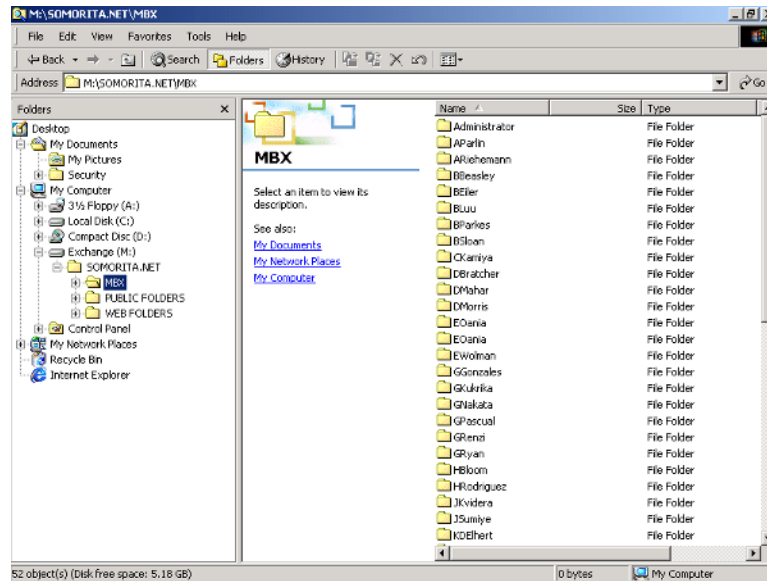
One of the new features of Exchange 2000 is the Exchange Installable File System (ExIFS), which allows mailboxes or any public folder tree to be accessed as if it were another network resource. When the Exchange information store starts, it also starts a kernel mode device driver called EXIFS.SYS, which interacts with the information store and allows access to the public folders and mailboxes.

From the server console, you can see a new drive letter (the M: drive by default). Figure 1.2 shows the M:\ drive and its root folder, SOMORITA.NET. Exchange takes SOMORITA.NET from the default recipient policy's default SMTP address. Under SOMORITA.NET, you can see the MBX folder and the Public Folders that allow access to the mailboxes and public folders.

The M: drive is available only from the Exchange server console unless it is shared. Once shared, any user can access any mailbox or public folder to which they have permissions.

---

**WARNING** Do not back up your mailboxes and public folder stores using the M: drive. Make sure that any file-based virus-scanning software that might be on the Exchange server excludes the M: drive entirely.

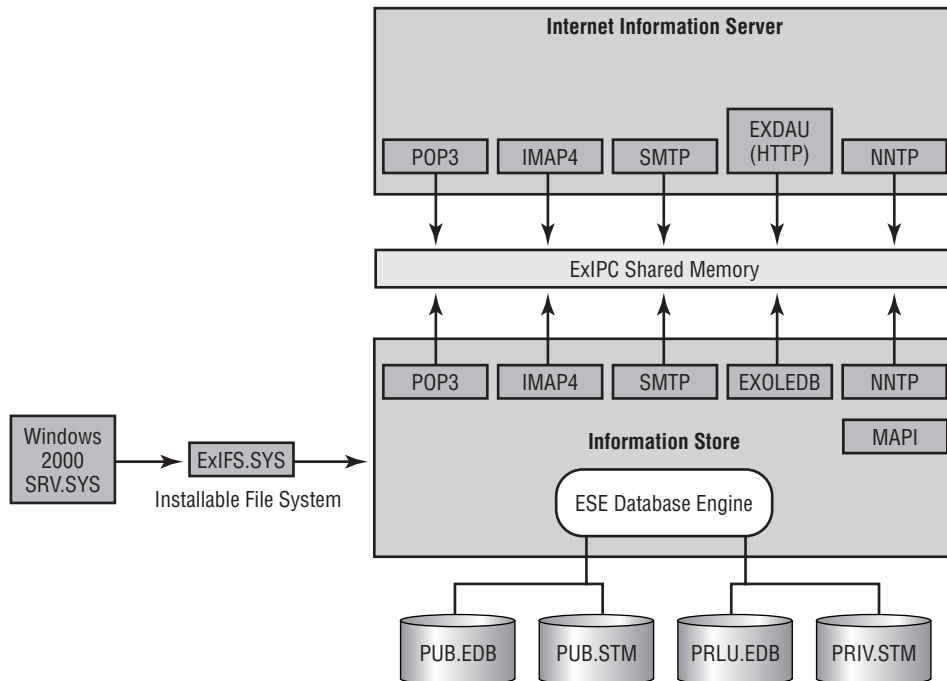
**Figure 1.2** The M: drive with a mailbox open

The ExIFS feature may be useful if you want users to be able to access their mailboxes or public folders through the file system. OWA and the Web storage system use ExIFS to access data stored in the Exchange mailbox and public folder stores.

### Internet Information Server and Exchange 2000

Internet Information Server (IIS) 5 plays an important role in the accessing and storing data in Exchange 2000. All Internet protocol support is now handled by IIS rather than by Exchange 2000 components. Figure 1.3 shows a basic architectural diagram of IIS and the Exchange 2000 information store.

All communication for POP3, IMAP4, SMTP, HTTP, and NNTP is now handled by IIS rather than being integrated into other Exchange components. IIS receives Internet protocol requests and messages, and passes these on to the information store. In order to achieve optimal performance, the Exchange developers implemented a shared memory layer between IIS and the information store called the Exchange Inter-Process Communication (ExIPC) layer. This layer is also referred to as EPOXY because it's the glue that holds the information store and IIS together, and thus the ExIPC DLL name is EXPOXY.DLL.

**Figure 1.3** IIS and Exchange 2000 information store interaction

Essentially, ExIPC is nothing more than an area of memory that the two processes share for queuing data and requests between them. Since it is shared memory, data and requests are transferred quickly and efficiently.

### The Advanced Queuing Engine

Perhaps one of the most dramatic changes between Exchange 5.5 and Exchange 2000 is the change to the message transport architecture. With Exchange 2000, all message transfer is the responsibility of the Message Transport System of which the Advanced Queuing Engine is a part. One of the design goals for the Exchange 2000 message transport system was to ensure that all messages were processed exactly the same. To that end, all messages are delivered through the Advanced Queuing Engine—even those that are destined for local delivery.

To do this without affecting performance and scalability is something of a monumental task. Further, all message transport in a native Exchange 2000 organization is via SMTP rather than RPC, so all Exchange 2000 servers must have the capability to transfer SMTP messages between servers in the same routing group. The Exchange 5.5 MTA used RPCs to transfer messages between servers in the same site.

In 1996, when Exchange 5.5 was released, Microsoft had three separate teams of developers working with SMTP: the Exchange team, the IIS team, and the Microsoft Commercial Internet System team. When Windows 2000 was being developed, Microsoft decided to combine these three teams into one group that would develop a single SMTP transport system to be used by all Microsoft components requiring SMTP transport.

---

**NOTE** All messages including those destined for local delivery are handled by the Advanced Queuing Engine.

### Message Transport Components

The IIS SMTP component is required prior to the installation of Exchange 2000. When Exchange 2000 is installed, it enhances (not replaces) several of the existing SMTP components so that they can work with Exchange 2000 more effectively. The SMTP transport components include:

**ExIPC** Provides the queuing layer that transfers message header information quickly and efficiently between IIS and the information store.

**Advanced Queuing Engine** Creates and manages the queues through which a message passes when it is being delivered. These queues include per-domain queues, the pre-categorization queue, the post-categorization queue, and the local delivery queue.

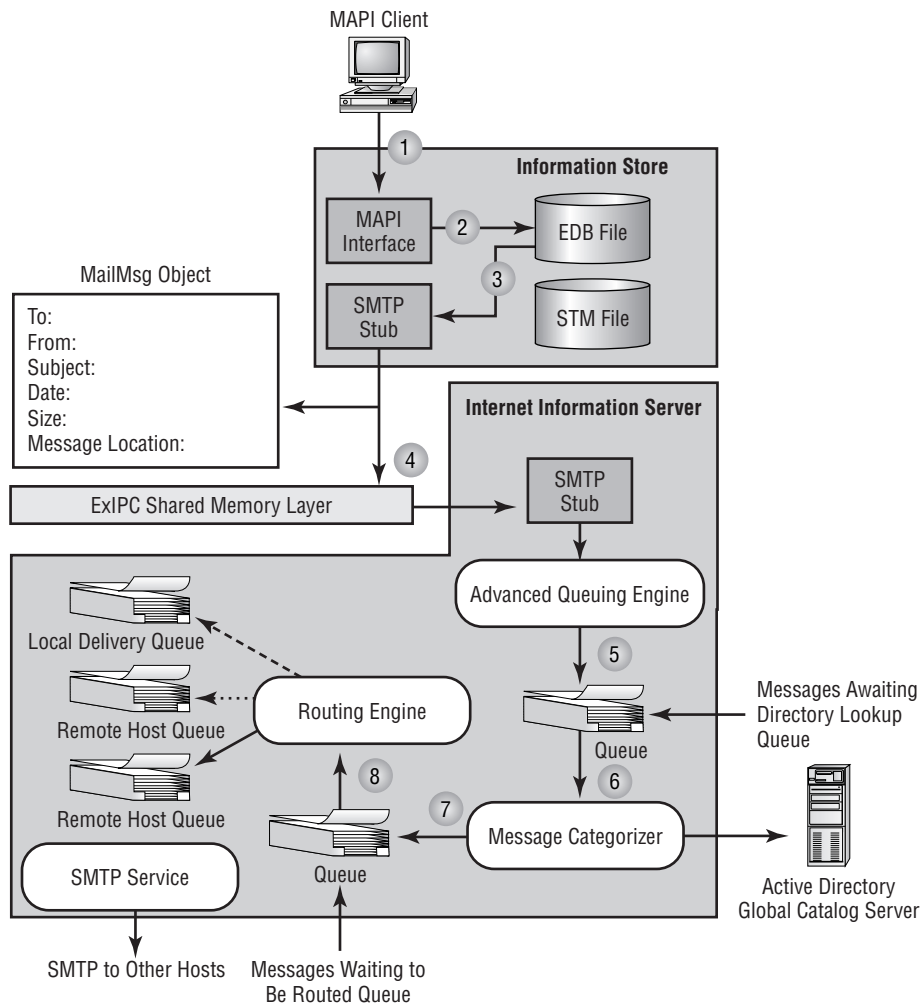
**Message Categorizer** Provides features specific to Exchange, such as checking recipient home servers, checking recipient limits, checking sender limits, and expanding distribution lists. This is an enhancement to the Advanced Queuing Engine. The IIS SMTP component has a basic message categorizer (`cat.d11`) that is disabled by default. When Exchange 2000 is installed, the Exchange categorizer (`phatcat.d11`) replaces the IIS categorizer.

**Routing Engine** Maintains the Link State Table, which is used by the Advanced Queuing Engine to determine the “next hop” through which a message needs to be routed. The Routing Engine also maintains information about whether or not a link is currently available.

**SMTP Service** Handles transmission of messages between hosts using the SMTP protocol.

When a message is transferred to the Advanced Queuing Engine, each component has specific functions that it performs to move the message to its next hop. Figure 1.4 shows a basic diagram of the Advanced Queuing Engine.

**Figure 1.4** SMTP Advanced Queuing Engine



If we follow the message through its path as it travels through the information store and Advanced Queuing Engine, it looks something like this:

1. A MAPI client submits a message through the information store's MAPI interface.
2. The information store determines that the message is a MAPI message and stores the entire message in the EDB portion of the mailbox store.

3. The information store creates an object that represents the message called the *MailMsg* object (also called the *IMsg* or *IMailMsg* object). This object is merely a small chunk of memory that identifies information such as the To, From, Subject, Date, Size, and other message properties, as well as where the actual message content is stored. In this case, the message content is stored in the EDB portion of the mailbox store. Only the *MailMsg* object, not the entire message content, is passed to the SMTP memory stub in the information store. The SMTP memory stub is a queuing location provided by the ExIPC queuing layer between IIS and the information store.
4. The information store's SMTP stub passes the *MailMsg* object through the ExIPC shared memory layer to the SMTP stub in IIS.
5. The *MailMsg* object is passed to the Advanced Queuing Engine, which stores the message in the Messages Awaiting Directory Lookup queue (Microsoft also refers to this queue as the Pre-Categorizer queue). You can see messages in this queue by using the Exchange System Manager and viewing the queues in the SMTP virtual server.
6. The *MailMsg* object proceeds to the Message Categorizer component, which takes message information—such as the sender, recipient, and size—and performs AD queries (to a Global Catalog server) to determine if the message exceeds the sender's or recipient's limits. Also at this point, gateway and routing restrictions are determined. If the message is sent to a distribution list, the Message Categorizer also expands the distribution list. If the message is being sent to both external and internal recipients, the Message Categorizer performs a bifurcation of the message (two or more copies are created) so that an RTF copy is sent to internal recipients and a MIME copy is sent to external recipients.

---

**TIP** When the Message Categorizer component is performing directory look-ups, connectivity to the Global Catalog server is critical. Any location that contains an Exchange 2000 server should also have a local Global Catalog server.

7. The *MailMsg* object is placed in the Messages Waiting To Be Routed queue (I also refer to this queue as the Pre-Routing queue).
8. The *MailMsg* object is handed off to the Routing Engine, which examines the destination domain or server and compares the destination with routes that are available in the Link State Table. If the message is for a local recipient, the *MailMsg* object is placed in the local delivery queue and the object is passed back to ExIPC. If the message is for remote delivery, the message is placed in the appropriate outgoing queue, and the SMTP service delivers the message off of the server. If the message is to be delivered by the message transfer agent (MTA) to Exchange 5.5 server

or to an X.400 connection, the message is routed back to the local store and placed in the MTA mailbox's MTS-OUT folder. Only when the message delivery to a remote host begins are the actual contents of the message moved out of the information store.

---

**NOTE** Only the MailMsg object—not the entire message—is passed through the Advanced Queuing Engine. The message content is moved only when the message is ready to be delivered to another server or store.

There are slight variations on this message routing process for inbound messages, but the process is essentially the same.

## Event Sinks

If all messages, regardless of whether or not they are destined for local delivery, are routed through the exact same message routing components, you might think this would be a good place to handle other types of message processing needs. If you thought this, you are not alone. The Exchange developers have introduced the concept of event sinks to Exchange 2000. An *event sink* is a small program that runs when a specific type of event occurs, such as a message arrival or the completion of categorization. In fact, the Exchange 2000 extensions to the IIS SMTP service are implemented as event sinks.

### Types of Event Sinks

There are three major categories of event sinks: information store events, transport events, and protocol events. When dealing with the Message Transport system, we are concerned mostly with the protocol and transport events.

*Protocol events* are used to extend SMTP functionality by enhancing or providing additional SMTP command verbs. Possible uses include rejecting all messages from domains that do not have a reverse lookup record, changing the behavior of an existing SMTP command verb, or adding a custom SMTP command verb.

*Transport events* can be used when the message is passing through the Message Transport system. Uses include content inspection, adding message disclaimers, anti-spam features, or message compression. Another use might be a virus-scanning service, but the Exchange 2000 antivirus API provides much better virus-scanning performance. Transport events can be fired at the following points:

- When a message is submitted to the Message Transport system (inbound from the information store or from the SMTP service)
- When a message is placed in the Pre-Categorizer queue

- When a message is in the categorizer
- When a message is in the Pre-Routing (Post-Categorizer) queue
- When a message is being processed by the Routing Engine

### Writing an Event Sink

The keys to writing a successful event sink are speed and accuracy. The accuracy part comes with comprehensive testing. However, the speed part comes with your choice of a programming language (and of course, writing efficient code). Development platforms include any programming language that is compatible with the Component Object Model (COM), including VBScript, JavaScript, Visual Basic, C, and C++.

If the event sink you want to develop will fire only for a select few messages an hour, then you can use Visual Basic, VBScript, or JavaScript. However, if your event sink will fire for all messages being processed, then you should use C or C++ to ensure maximum performance.

### Exchange 2000 Connectors

By default, SMTP is used between Exchange servers in the same routing group. There is no need to set up any special configuration since an SMTP virtual server is configured by default on all Exchange 2000 servers. However, like connectivity to other Exchange 5.5 sites, connectivity to other Exchange 2000 routing groups requires a messaging connector. Exchange 2000 offers these options:

**Routing Group Connector** The preferred method of communication between routing groups. It uses SMTP to connect between routing groups unless it is being used to communicate between an Exchange 2000 routing group and an Exchange 5.5 site, in which case it uses RPC. The Routing Group Connector is sort of like the old Exchange 5.5 site connector on steroids and using SMTP. It uses multiple bridgehead servers, can be scheduled, and can defer large messages to a time when WAN connectivity is not at a premium.

**SMTP Connector** Typically used when you want to focus outbound SMTP connectivity to the Internet. If the SMTP Connector is not installed, then *any* Exchange 2000 server can deliver an SMTP message to the Internet. Once an SMTP Connector is installed, an SMTP address space is added to the Link State Table, indicating that all SMTP messages not destined for another Exchange 2000 server in the organization should be routed to this connector, but only if the address space is set to “\*”. The SMTP Connector can also be used to connect to Exchange 2000 routing groups.

**Message transfer agent (MTA)** Used to communicate with other Exchange 2000 routing groups using X.400 and with foreign X.400 systems such as the U.S. Department of Defense’s Defense Messaging System (DMS).

Other connectivity options that are part of Exchange 2000 include:

- Connector to Microsoft Mail
- Connector to Lotus Notes
- Connector to Lotus cc:Mail
- Connector to Novell GroupWise

Though Exchange 5.5 shipped with the connectors for PROFS- and SNADS-based systems, these are not included with Exchange 2000. If you require these connectors, you must keep your organization in Mixed mode and continue to operate an Exchange 5.5 server. If you are starting from scratch, you must install the Exchange 5.5 server first and then install your Exchange 2000 servers into the Exchange 5.5 organization.

## **Exchange 2000 System Attendant**

The Exchange System Attendant (MAD.EXE) is essentially the general manager of the Exchange server. It is the first Exchange service that starts and the last one that shuts down. While a novice might actually think that this service performs few, if any, useful functions, it actually is responsible for a lot of odd yet important jobs. Some of the tasks that the System Attendant runs include:

- Performing offline address book generation.
- Running the DS2MB (Directory Service to Metabase) update process to keep the IIS Metabase in sync with the information in Active Directory.
- Generating proxy addresses for X.400, SMTP, and other address types based on the defined Exchange 2000 recipient policies.
- Emulating the Exchange 5.5 directory service through a process called DSProxy for MAPI clients prior to Outlook 2000 that cannot receive referrals.
- Passing referrals for Outlook 2000 and later clients that need to be referred to a Global Catalog server for querying address information.
- Running the Recipient Update Service to make sure that AD objects are included in the appropriate address lists.
- Running the DSAccess cache, which caches information about AD objects. This cache is available for Exchange 2000 to query rather than querying the AD directly for each lookup request.
- Inserting data into and managing the message tracking logs.

## Exchange 2000 Modes

Like Windows 2000 Active Directory, Exchange 2000 has two modes in which the organization can operate: Mixed mode and Native mode. AD Native mode and Exchange 2000 Native mode have no effect on each other; they are completely independent.

By default, the organization is in Mixed mode, which allows Exchange 2000 to interoperate with Exchange 5.5 servers. Several limitations are imposed on an Exchange 2000 organization that is operating in Mixed mode, including:

- Windows 2000 Administrative groups are mapped directly to the Exchange 5.5 site architecture.
- Routing group membership can consist only of the servers that are in the administrative group containing that routing group.
- Exchange 2000 servers cannot be moved between routing groups.
- RPCs are used between Exchange 2000 servers and Exchange 5.5 servers.

To switch the organization to Native mode, check that all Exchange 5.5 servers have been upgraded or removed from service, and remove all ADCs and site replication services (SRSs). (Make sure to remove the SRS from the Tools container in Exchange System Manager.) Then display the Exchange organization's properties using Exchange System Manager and click the Change Mode button.

---

**WARNING** Changing to Exchange 2000 Native mode cannot be reversed.

Once you are in Native mode, you will have a little more flexibility than you do in Mixed mode. Some of its features include:

- Each administrative group can have multiple routing groups or no routing groups.
- A routing group can contain servers from any administrative group.
- Servers can be moved between routing groups.
- SMTP is used as the default message transport protocol between all Exchange 2000 servers.

## Picking the Right Edition of Exchange

Now that you understand some of the basics of Exchange 2000 and how it differs from Exchange 5.5, it's important that you understand the differences between the two editions of Exchange 2000: Exchange 2000 Server and Exchange 2000 Enterprise Server.

You must pick the right version to meet the needs of your organization. Table 1.2 lists available features and which edition of Exchange 2000 provides them.

**Table 1.2** Features with Exchange 2000 Server and Exchange 2000 Enterprise Server

Feature	Exchange 2000 Server?	Exchange 2000 Enterprise Server?
Active/Active clustering (Clustering can be either Active/Active or Active/Passive depending on the number of nodes.)		✓
Active Directory integration	✓	✓
Chat services		✓
Content indexing and searching	✓	✓
Database size larger than 16GB		✓
Exchange Installable File System (ExIFS)	✓	✓
Exchange policies	✓	✓
Front-end/back-end configuration	Can only function as a back-end server, but can be "front-ended" by an Exchange 2000 Enterprise server.	Can function as a front-end server or a back-end server.
Instant Messaging	✓	✓
Multiple mailbox stores		✓
Multiple storage groups		✓
Routing Group Connectors	✓	✓
SMTP Connector	✓	✓

**Table 1.2** Features with Exchange 2000 Server and Exchange 2000 Enterprise Server (continued)

Feature	Exchange 2000 Server?	Exchange 2000 Enterprise Server?
Web storage system	✓	✓
Windows 2000 security	✓	✓
Workflow Designer for Exchange 2000	✓	✓
X.400 Connector		✓

**NOTE** Exchange 2000 Conference Server is not included with either edition of Exchange 2000; it is a separately licensed product. Exchange 2000 Conference Server requires at least one Exchange 2000 server and one Windows 2000 domain controller.

## Upgrading between Editions

You can easily upgrade from Exchange 2000 Server to Enterprise by simply running the Exchange 2000 Enterprise Server Setup program and choosing the Reinstall option.

However, you cannot “downgrade” from the Exchange 2000 Enterprise Server version to Exchange 2000 Server. If you must do this, consider installing an additional server using Exchange 2000 Server and moving the mailboxes over to that new server.

**TIP** How can you tell which edition you have? Check under the server’s Protocols container and see if you have an X.400 container. If so, that server is an Enterprise Server. You can also review your event logs and look for event 1217 from the MExchangeIS Mailbox Store. This indicates that the mailbox store has unlimited capacity.

## Storage Considerations

Exchange 2000 is far more scalable than its predecessors. Some Exchange 2000 administrators are reporting that they are now supporting nearly a terabyte of mailbox data on single servers. As the number of mailboxes on servers increases, you need to plan carefully

to make sure that you're keeping a manageable amount of storage and that the mailbox data is backed up—and backed up in a reasonable amount of time.

## Storage Limits

During the first Exchange 4 beta, I discovered storage limits, a feature sadly missing from other messaging platforms I had supported in the past. I asked a Microsoft consultant what the recommended limit was for mailboxes. His reply? 50MB per mailbox. As a cc:Mail administrator, I would have panicked if my entire database approached 50MB!

The changing nature of messaging systems has launched the need for more and more storage. E-mail messages are more and more complex with the advent of RTF and HTML formatting. And users love to send attachments. A typical Microsoft Word document with a few pages of text can easily be 40KB without any graphics. Several Exchange administrators report to me that they see 100MB Microsoft PowerPoint attachments all the time. The increasing popularity of digital signatures also increases the average message size. I know numerous people who include their company logos in their personal message signatures—and even one person who includes his picture (every message he sends is automatically 72KB!).

Many Exchange installations have embraced third-party add-on products, which include fax gateways, work-flow solutions, and voicemail integration—all potential space hogs. A standard two-page fax received into my mailbox is about 80KB, and a typical one-minute voicemail message is about 125KB.

Even simple, innocuous looking forms can put a burden on storage requirements. The “While You Were Out” form that is included with Outlook creates a message that is nearly 80KB. Messages that contain a form that is not in the organizational forms library can be 100KB or greater simply because the form is stored with the message.

Additional features of Outlook such as calendaring, contacts, journaling, notes, and tasks continue to drive the need for larger mailbox storage limits. A quick glance at my own Exchange mailbox shows that I have over 5MB of data in my Contacts folder, nearly 2MB of data in my Calendar folder, and 10MB in my Journal folder.

---

**WARNING** Outlook's Journal folder can easily accumulate many megabytes of data if the journaling feature is turned on. Advise your users not to turn on this feature unless they require it. Alternatively, configure a Windows NT policy that disables this feature.

### Exchange@Work: Setting Storage Limits

What does the typical organization do with respect to storage limits? Well, the only consensus is that there is no consensus. I still know a few organizations that require their users to use personal folders (PST files); all received e-mail is downloaded to the user's PST file storage in their personal profile or home directory.

I also know of a few companies that do not set any limits other than the Prohibit Send And Receive limit, but they use the Mailbox Manager (available with Exchange 2000 SP1) to automatically clean out all Sent items and Inbox items that are older than 60 or 90 days.

Still other folks set mailbox limits fairly low (20MB Issue Warning limit, 25MB Prohibit Send limit), but require that the users keep their own mailboxes clean. Some of these will configure Outlook's auto-archive feature to automatically move messages older than 30 days to a PST file.

I do know one administrator who purges older messages in users' mailboxes with the `exmerge.exe` utility from the Exchange 2000 CD-ROM (found in the `Support\Utilities\I386\exmerge` directory). She archives messages older than 120 days from the mailboxes and then backs up on tape the PST files that are created.

Obviously there are numerous approaches to handling storage issues (even more options are discussed in Chapter 4!), but the majority of administrators I know set their Issue Warning and Prohibit Send limits between 40MB and 75MB, and then override these limits for individual mailboxes. Keep in mind that when setting limits on mailbox storage, you have to have balanced the need to have a manageable amount of mailbox data with making sure the system remains usable to end users. Applying a policy that gives users only 5MB or 10MB of server-based storage will certainly help reduce your server-based storage requirements, but it may not allow the user anything more than a few days' worth of e-mail on the server.

---

**TIP** Even if you set no warning or send limits on mailboxes, configure Prohibit Send And Receive limits to prevent a mailbox from being spammed and possibly causing the server to run out of disk space. On several occasions, I have seen a single mailbox cause an entire server to shut down.

## Server-Based Storage Versus Local Storage

One of the age-old questions that Exchange administrators have been facing since Exchange 4 was released in 1996 is “Where should users store their mail?” Both the original Exchange client as well as all versions of Outlook provide a storage option called a personal store (PST file). These clients give the user the option of automatically downloading all mail that was sent to the user’s mailbox on the server to the local PST file.

The PST file is a simple B-tree database that can be located on either the user’s local hard disk or server-based home directory. Yet based on the experiences of many Exchange administrators as well as my own, I strongly recommend that the primary message storage medium be the Exchange information store, not the local PST file. Table 1.3 compares PST-file storage and Exchange server-based storage.

**Table 1.3** PST-Based Mail Storage vs. Server-Based Mail Storage

PST-Based Mail	Server-Based Mail
Reduced storage capacity is required on server.	Single instance storage is maintained.
PST files can be password protected, but the password can be cracked easily.	Mailbox security is centrally controlled and audited by the Exchange administrator.
Locally stored PST files may not be backed up if the local machine’s hard disk is not being backed up.	Backups are centralized and administrator controlled.
A PST file may consume twice as much or more disk space as server-based storage because messages are stored twice in the PST file—once in RTF and once in plaintext—to maintain compatibility with older clients, and all forms are stored with each message.	Message storage is limited by the administrator.
Maximum PST file size is 2GB or 64,000 entries per folder. When the file size nears 2GB, it becomes corrupt, and you may not be able to recover any information from it.	
PST files are subject to database corruption.	

**Table 1.3** PST-Based Mail Storage vs. Server-Based Mail Storage (*continued*)

PST-Based Mail	Server-Based Mail
PST cannot be shared or accessed by multiple systems.	
PST files can't be accessed from OWA.	

## Time Synchronization

When using Windows NT 4 and Exchange Server 5.5, time synchronization was not required. Even with Windows 2000, AD replicates and synchronizes directory entries, not based on time, but by using *update sequence numbers (USNs)*. Exchange 2000 public folder replication is still based on a list of changes to each item called the *predecessor change list*. Time values for changes are used only in the event of public folder design change conflicts.

Based on this information, you might believe that time synchronization is not necessary. However, Windows 2000–based networks use Kerberos authentication, which requires that computers have their time synchronized to within five minutes. This means that all Windows 2000–based computers that are part of an AD forest must have their time synchronized to a common time source.

### Synchronizing Windows 2000 Computers

Unlike previous version of Windows, Windows 2000 includes a service (the *W32Time service*) that ensures all Windows 2000 computers in an organization use a common time. Understanding how the time is synchronized is helpful, especially if you want to synchronize your network's time to an outside source.

In order to maintain proper time synchronization between Windows 2000 computers, the following hierarchy is used:

- Windows 2000 Professional computers use the domain controller that authenticated their login as their time source.
- Windows 2000 member servers use the domain controller that authenticated their login as their time source.
- Windows 2000 domain controllers use the PDC (primary domain controller) Flexible Single Master of Operations (FSMO) as their time source.

- Windows 2000 PDC FSMOs in child domains use the Windows 2000 PDC FSMO in their parent domain as their time source.
- The Windows 2000 PDC FSMO in the root domain of the forest is considered the authoritative time source for the entire forest.

Since the root domain's PDC FSMO is the master time source for all Windows 2000 computers in the forest, it should be configured to use an external time source. The easiest way to do this is simply to keep this server set to the correct time. Though not terribly accurate, you can call your local time and temperature telephone number periodically and set this server's time accordingly. You can get more accurate time information from the United States Naval Observatory (USNO) at [tycho.usno.navy.mil/what.html](http://tycho.usno.navy.mil/what.html).

If you have Internet connectivity, you can automate this process by setting the preferred time source for the root domain's PDC FSMO. If you want to set the server to use the USNO SNTP (*Simple Network Time Protocol*) time server, type the following at the command prompt:

```
NET TIME /SETSNTP:TOCK.USNO.NAVY.MIL
```

---

**WARNING** Setting the SNTP server name on Windows 2000 member servers and Windows 2000 Professional computers will appear to work, but it won't actually set the time. Only Windows 2000 PDC FSMO computers can use an external time source using the NET TIME command.

In order for this command to work, the PDC FSMO must have Internet connectivity. If you have a firewall, SNTP must be allowed through it. SNTP uses UDP port 123.

---

**NOTE** For a list of NTP and SNTP servers around the United States, visit [tycho.usno.navy.mil/ntp.html](http://tycho.usno.navy.mil/ntp.html).

## Synchronizing Non-Windows 2000 Computers

If you have Windows NT 4, Me, 98, 95, or (shudder) 3.x computers on your network, you may also want to synchronize their system clocks. While this is not required for authentication, it is a good practice to make sure all the clients on the network have the same time. The simplest way to this is from the command prompt. Pick a server on your network as the master time server for these computers (in my example it is called HNLDC01) and type the following at the command prompt:

```
NET TIME \\HNLDC01 /SET /Y
```

This works fine for Windows 9x and 3.x based computers, but you must be a member of either the local Administrators, Power Users, or Server Operators group in order to make this change on Windows NT 4 computers. If you don't want to assign a user membership in one of these groups, you can simply assign them the local right "Change the system time."

Since some computer's time settings drift as much as 10 minutes per week (as one of my notebook computers does), in order to keep the time synchronized automatically, consider placing the NET TIME command in the logon script.

### Automatically Synchronizing Windows NT 4 Computers

The Windows NT Server 4 Resource Kit has a tool called the Time Synchronization Service. At certain intervals, this tool gets the current time from a time source that you have defined. The time service tool can use a TCP/IP network or a modem to contact a time source.

There are two types of time sources. A *primary time source* is usually an external and highly accurate time source located somewhere like the Naval Observatory or the National Institute of Standards and Technology (NIST). A *secondary time source* is set from a primary time source.

When I use the Time Synchronization Service, I pick a server on my network that has a connection to the Internet to designate as my primary server. Then I install the time server software and edit the `timeserv.ini` file to set the time source type to INTERNET and to set the default period to 12. This points my server to the US Naval Observatory to get time updates every 12 hours. When I start the time service, an optional parameter in the `timeserv.ini` file tells the time service to log events to the Application event log.

On all other servers on the network, I install the time-server software and edit the `timeserv.ini` file to designate those servers as SECONDARY. I designate the primary server that I have just installed as the PrimarySource and set the period to 12 hours in the `timeserv.ini`. Every 12 hours, these servers will contact the primary server and get the updated time.

## Standardization

One of the most important and useful things you can do for your organizations is establish standards. Perhaps one of the biggest mistakes I saw for organizations implementing Exchange 5.5 was a lack of common standards for hardware, the Exchange organization, site names, alias names, display names, SMTP addresses, group names, and server names.

To say that a lack of planning and standardization will be a stumbling block for the implementation of Exchange 2000 would be the biggest understatement I have ever made. Without planning and standardization, Exchange 2000 (and Active Directory) implementations don't hit stumbling blocks—they hit brick walls.

## Active Directory

Windows 2000 Active Directory is the single biggest factor affecting the successful implementation of Exchange 2000. Exchange 2000 uses AD to store all Exchange 2000 configuration information, mailbox attributes, connector information, administrative permissions, and routing information, just to name a few. Some important facts and factors to keep in mind when planning your organization's Active Directory include:

- A single Exchange 2000 organization cannot span multiple AD forests.
- Only one Exchange 2000 organization can exist in any AD forest.
- The AD schema is replicated to *all* domain controllers in the entire forest.
- Create a single AD forest that will contain all Exchange 2000 servers and mailboxes.
- Use a common DNS namespace for all servers and workstations.

### **Exchange@Work: Three Forests, One Company**

XYZ Corporation was an early adopter of Windows 2000. They installed three separate Windows 2000 forests; each forest contained a single domain. Each of these domains was in a different physical location, and the local administrators performed most day-to-day administrative work.

When it came time to upgrade their Exchange 5.5 servers, they were disappointed to learn that they were going to have to migrate two of the Windows 2000 domains into a common forest. While there are tools available to help migrate accounts to a new domain, this is not the same as moving the domain into a new forest. Many hours were spent preparing for and carrying out this migration. The ultimate forest design consisted of a single Windows 2000 domain with users from the different locations split into multiple organizational units.

If XYZ Corp. had initially created a single AD forest, they would have saved themselves many hours of additional work reconfiguring their AD domain structure.

---

**NOTE** More information on Active Directory is found in Chapter 2. For in-depth information on Active Directory, consult *Mastering Active Directory* by Robert R. King (Sybex, 2000) or *Mastering Windows 2000 Server* by Mark Minasi (Sybex, 2000).

### Username

Active Directory becomes the global directory service for your entire organization. Every user account you create in AD is visible to the entire organization, including the information that you input for that user. The standard you develop for creating user accounts should include:

- Alias name
- Display name appearance
- Additional information that's entered into Active Directory, such as SMTP addresses, mailing address, title, department, manager, etc.

**Alias Names** In most organizations, the alias name is exactly the same as the Windows user account name. There are a number of possible standards you could choose to implement:

NKarasuda	First name initial followed by last name
KeithS	First name followed by first initial of the last name
JWM	Initials
US2632	Employee number or job code

In any organization with more than a few dozen employees, you are most definitely going to run into problems with duplicates. Make sure that your standard allows for duplicate names and provides a standard formula for conflict resolution that creates unique names.

---

**TIP** Some organizations are now choosing a separate e-mail address from their Windows user account. The reason is simple security. If you choose the same Exchange alias (and thus the SMTP address) as your Windows user account, you are giving a potential hacker half of the equation (hopefully, given strong passwords, the easier half) for accessing your network.

**Display Names** The display name is one of the most important directory attributes to keep consistent. Why? Because this attribute is the one that users see when they display the global address list and that users of foreign systems see when they receive a message from

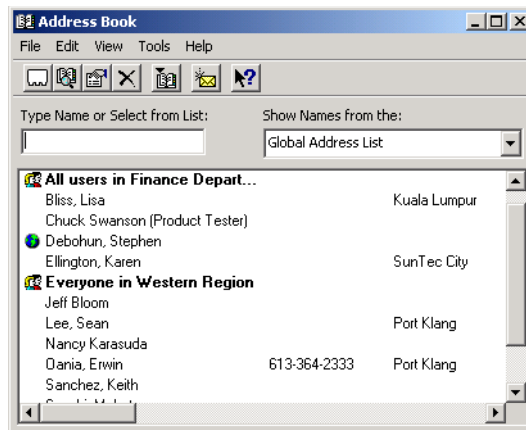
your users. Many organizations that have multiple domains and administrators have unfortunately seen each administrator creating their own standards for how a display name should look. And something as simple as commas being placed in different places or displaying the first name before the last name can make finding a mailbox difficult.

When choosing a standard for the display name, keep in mind that the display name is the user's relative distinguished name (RDN) and is designated by the "CN" value in the distinguished name (DN). Creating complex DNs may make it difficult to build custom Active Directory applications. Here are a number of standards that I have seen implemented:

Keith Sanchez	First name, last name (the default display name)
Sanchez, Keith	Last name, comma, first name
Sanchez Keith SFC 30 SigBn Ft Shafter	Last name, first name, title/rank, department, user location (a popular structure in the military)
Sanchez, Keith (DFW – Systems Engineer)	Last name, comma, first name, user location, job title (a popular structure among larger organizations)
Sanchez, Keith M.D.	Last name, comma, first name, title

Figure 1.5 shows the global address list as it appears in an organization that has not established any standards for display names. Note that there is no consistency in how the names are displayed.

**Figure 1.5** Global address list for an organization with no display naming standards



**SMTP Addresses** Depending on your organization, standards in SMTP addresses may also be important. I have worked in a number of organizations that were all over the board with respect to what the SMTP address actually is. Again here, a predetermined sequence of naming rules is useful for resolving conflicts that arise. Here are some examples of possible SMTP address standards:

GKukrika@somorita.com	First initial, last name (the most common SMTP naming standard)
GogaK@somorita.com	First name, last initial
Goga.Kukrika@somorita.com	First name, period, last name

---

**NOTE** SMTP addresses cannot contain spaces or commas. They are not case sensitive.

### Group Names

Active Directory enables you to create two types of groups. Either of these groups can be “mail-enabled” for use with Exchange 2000. These group types are:

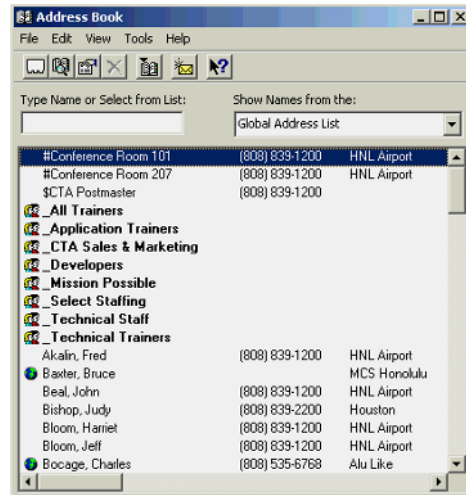
- A *security group* allows permissions and user rights to be assigned to the membership of the group by assigning the permission or right to the group. Security groups are also used for assigning public folder permissions.
- A *distribution group* is used only for sending e-mail messages. Permissions and rights cannot be assigned to a distribution group.

---

**NOTE** Don’t confuse group types (security or distribution) with group scopes (Domain Local, Global, or Universal). These are discussed in more detail in Chapter 2.

Any group that is “mail-enabled” will appear in the Exchange global address list and can be used as a distribution list. The Exchange global address list does not differentiate distribution groups from security groups. When creating display names for these groups, a good practice to consider is using some type of special character to control where these groups display in the address list.

Figure 1.6 shows the Outlook address book view of an organization’s global address list. Note that the mail-enabled security groups all have an underscore character (\_) in front of them. The conference room resources contain a pound symbol (#), and system accounts have a dollar sign (\$) in front of them. These special characters cause the different types of objects to be sorted together.

**Figure 1.6** Outlook address book showing security groups sorted together

### Server Names

Windows computer names should be standardized to avoid conflicts. You should develop a standard that will be meaningful to your organization and prevent conflicts. I have found administrators who are fond of using server names such as SERVER1 or using the user's name who works at a particular workstation. However, SERVER1 may be duplicated at another location. Users often move to new computers, or the computer is replaced with an upgrade model.

You should develop a standard that is independent of the user's name and does not allow for duplicates between offices or locations. Additionally, the standard should include a location code to identify the location of the computer (I am fond of the city's airport code), as well as an identifier showing what type of server it is. Below are some examples of naming that I have found to be useful and versatile:

Server Name	Explanation
SFODC01	San Francisco domain controller 01
HNLEXCH02	Honolulu Exchange server 02
KULSQL01	Kuala Lumpur SQL server 01
SINPROX03	Singapore proxy server 03
LJUISA02	Ljubljana ISA server 02
NRTWS0644	Tokyo workstation 0644

---

**WARNING** Do not use spaces or the underscore ( \_ ) character in server names.

## Server Hardware Considerations

Do you want 99.9% uptime on your Exchange 2000 servers? We all do, but most people don't realize that 99.9% uptime allows you only about nine hours of downtime per year. 99% uptime gives you about 3½ days per year that you can actually be down. Three things will help move you toward that 99.9% uptime category:

- Good operational procedures.
- A reliable, well designed hardware platform.
- A good operational definition of “downtime” and “uptime.” Make sure that you note the difference between scheduled/planned outages and system failure. There is no shame in having 10–15 hours of planned downtime if it means you have zero unplanned downtime.

Though most of this book is dedicated to helping you establish good operational procedures, this section will help you make the right decisions when choosing server hardware for Exchange 2000.

### Choosing Hardware Components

Choosing the right brand of server hardware and the components for that hardware will help you achieve significantly better uptime. I believe the most important decision with respect to hardware choice is the choice of the hardware manufacturer. Further, I highly recommend that once you have chosen a manufacturer for your major components, you use that manufacturer for your minor components (disks, RAM, network adapters, SCSI adapters, tape drives, etc.) as well.

Sure, this may prove to be a little more costly in the short term. But any systems engineer who has tried to integrate components from different manufacturers will tell you that integration time for components from a single manufacturer is significantly less than the time it takes to build a server containing components from many different manufacturers. Further, if there are problems with any of the components, you have only a single vendor that you have to work with. So when making hardware decisions, pick a reputable hardware manufacturer and stick with them for all of your components.

### Preventing Disasters with Hardware

As I stated earlier, most of the disasters that I have helped recover recently have been due to the inexperience of the system administrator, or the problem has been compounded by incorrect user actions. However, the other system failures that I have had to help recover

in the past year have been a result of poor choices with respect to server, disk, and network hardware. These disasters have included a failed server that had run out of disk space, a disk failure on an unmirrored transaction log disk, debugging performance problems, and a bizarre case of Windows 2000 blue-screening a few times a week. This was one of the easiest problem-solving gigs of my career: The customer was running a Windows 2000 domain controller, DHCP, DNS, and Exchange 2000 on a single Pentium 700MHz system with 128MB of RAM. Their vendor swore that they had 1GB of RAM, but never bothered to physically check.

### **Disk Redundancy**

The system component that fails the most often is the hard disk. Sometimes you are lucky and a server can operate for many years without a single disk failure. Other times you can be remarkably unlucky (like one of my clients), and two of the three disks in a RAID 5 array will fail inside of one week.

When configuring a new server, one hard and fast rule you should live by is this: All hard disks should be configured with *redundancy*. This includes the operating system disk, log file disks, mailbox and public store disks, and swap file disk.

Recently, I reviewed a proposal from a large, international consulting firm that recommended disk redundancy for only the database disks and the operating system disk. The consultancy's logic was that if the log file disk failed, the data would still be intact on the other disks. This is true to a certain degree, but the server will still shut down. And how long will that server be offline before the disk is replaced? Data redundancy is certainly important, but keeping the server online and available to the users is just as vital.

### **Disk Space Requirements**

How much disk space should you plan for? And should you purchase the disk space immediately, or add the disk space as you need it? Typically, I recommend the following formulas for calculating how much disk space is required:

- Provide at least a 4GB system disk.
- The transaction log file should be able to store at least five days' worth of transactions without being purged.
- Each database disk should have free space that is at least the size of the largest database on the disk available. For example, if the largest database on the disk is 100GB, then there should be 100GB of free disk space in addition to the 100GB database.
- The disk that contains the SMTP \Mailroot\Queue directory should contain enough disk space to store at least 12 hours' worth of incoming mail. If you don't know how much this is, then plan for at least 2GB.

As for adding disk space as it is required, this is one recommendation I would *not* make. Purchase all the disk space you think you will need for a few years. For most servers, adding additional disk space means downtime and administrative overhead.

#### **Exchange@Work: Disk Space Usage**

ABC Corporation supports a site with four Exchange 2000 servers. Two of these servers have mailboxes, the third is dedicated to public folders, and the fourth is their bridgehead server (containing the fax connector and the SMTP Connector).

Each of the two mailbox servers has approximately 2300 mailboxes, most of which are accessed on a daily basis. The average user has a 65MB Send And Receive limit on their mailbox and a 50MB Warning limit. The largest mailbox store is nearly 95GB.

It's interesting to note that each of the mailbox servers generates 650 transaction log files per day, on average. When the nightly backup runs, the transaction log disk has about 3.5GB of transaction logs. The transaction log disk has approximately 16GB of usable disk space, so the server can run for about 4½ days without performing a normal or differential backup. Each volume that is used by a mailbox store is partitioned (on a storage area network) with 200GB of usable store. I think this is a great setup; it gives ABC a good safety net in a number of areas, including the transaction log disk and the mailbox stores.

---

**WARNING** Running out of disk space on the transaction log disk is one of the most common reasons for Exchange servers going offline.

#### **Tape Backup Hardware**

Today's tape backup hardware is very impressive. The choices of available backup technology include DLT tape arrays, auto-loader systems, and more. Many companies choose to perform centralized backups rather than installing backup devices on their Exchange servers. The advantage of centralized backups is that they require less administration, and you can purchase more sophisticated loader and tape library management systems if you are supporting one centralized system.

The disadvantage of centralized backup and restore systems is that they must back up the data across the network, which typically means that it takes longer. One of my clients implemented a centralized DLT backup. This centralized system could back up remote

Exchange data at a rate of about 3GB per hour. They found this rate to be unacceptable, so they moved this backup hardware into a storage area network that was attached to each of the Exchange servers locally. The backup rate improved to almost 25GB per hour. They later found out that the backup server was actually on a 10MB network, not a 100MB network. Once the backup server was moved to a 100MB switched network, the backup rate improved to almost 15GB per hour.

The points I try to follow when I recommend or purchase tape backup hardware include:

- The hardware must be able to perform a full (normal) backup of the all Exchange information stores each night and allow each store to complete at least one pass of nightly maintenance.
- The restoration time of the largest mailbox or store must fall well within the maximum continuous downtime limit specified by the service level agreement.

### **Uninterruptible Power and Power Supply Redundancy**

Though providing an uninterruptible power supply (UPS) for all server hardware seems like an obvious recommendation, it is one that is often overlooked. Uninterruptible power is essential for not only Exchange servers, but for all components in your computer room. This includes hubs, switches, and routers. Exchange should continue to operate normally and have access to the necessary domain controllers and Global Catalog servers after a power failure.

Even if you have a building power supply and generator, the UPS is still a good investment. One of my customers recently had a commercial power failure and found that the generator system took a full five seconds to cut over. Naturally, all of the servers were offline about five seconds before the generator kicked in.

Further, UPS monitoring should be configured so that all servers will shut themselves down if the commercial power is not restored in a timely fashion. Note that active Exchange servers can take significantly longer to shut down cleanly than other servers, like file or print servers. If you must commence an automated shutdown, make sure that your servers have enough time to completely shut down prior to the battery power being exhausted.

How much battery life should the UPS provide? Here in Hawaii, our typical power outage, however infrequent, usually lasts no more than 10 minutes. So I feel that 15 minutes of battery power should be sufficient. However, living in the U.S., I am lucky to have consistently reliable power. Many people around the world are not as fortunate. Overseas, I have experienced power outages between five minutes and two or three days. If your primary electrical power source is frequently unavailable for more than 15 minutes, consider generator backup solutions.

If you are running complex disk storage systems such as storage area networks or network appliances, an emergency generator or a UPS that will keep you up longer than 30 minutes is essential. Some of the new SAN (storage area network) solutions I have worked with are about as complex to shut down as a mainframe and can take upwards of 20 minutes for a controlled shutdown!

---

**TIP** Vendors that ship server class hardware today provide the ability to configure the server with multiple power supplies. Make sure that the server can run with a failed power supply. And learn how to monitor the power supplies so that you can determine if one has failed.

### Cold Standby Servers

One of my favorite things to do in an environment that is installing several Exchange servers is to make sure that they are all configured identically and to specify an additional server that has no specific function. This server has two purposes. The first and official job is to act as the cold standby server. If any hardware failure occurs to any of the other servers, I can simply move the disks over to the standby server and bring it back online.

The second purpose of the cold standby is to be used for disaster recovery practice and training. Disaster recovery practice and test data restores are essential for a healthy organization. Ideally, we hope that we will never have to restore data from tape or use our disaster recovery skills, but I promise you that your first disaster will go much more smoothly if you have some practice.

---

**TIP** Make sure that you have written (and hard-copy) instructions of the procedures for switching a server over to a cold standby in the event the production server fails. In one site, I developed meticulous documentation for this, but left it stored on one of the servers. When we got ready to test the switchover, we had to bring the original server back up in order to print the documentation.

### Hardware and Performance Tuning

Purchasing the right hardware and setting it up for disaster prevention is not enough; the hardware must be configured properly, and Exchange 2000 must be configured to use the hardware effectively. This includes which disks Exchange Server uses for mailbox and public folder stores and transaction logs as well as making sure the RAM on the server is configured properly.

### One Server, One Task

Want to avoid problems? Do not overlap tasks on the same machine. Exchange Server should be on a member server; it should not be installed on the same server as a Windows 2000 domain controller or Global Catalog server, nor should other application servers such as SQL Server, SMS components, or ISA Server be installed on the same machine.

### RAM Requirements

How much memory is enough for Exchange? How much memory does your server hold? Max it out, and you'll probably have enough. Seriously though, I recommend a starting point of 512MB of RAM. For servers that are going to support more than 300 mailboxes, plan to go to a minimum of 1GB of memory. For more than 500 to 1000 mailboxes, go ahead and max out the server at 4GB of RAM.

---

**NOTE** Exchange 2000 will attempt to allocate all of the free memory with the information store processes (`store.exe`) using the most memory. It is not unusual to see the `store.exe` consuming almost all of the memory on a server. If the server has 1GB of RAM, the store process may well have allocated over 800MB of RAM. This is normal and should not be a source of concern.

If memory is at a premium on your Exchange server, create fewer mailbox and public folder stores. Create all of your mailbox stores and public folder stores in a single storage group. Each mounted mailbox store allocates a minimum of 10MB of RAM. The first store in a storage group allocates even more memory. So with some innovative creation of mailbox stores in a single storage group, you can reduce your overall memory consumption.

If your server hardware has more than 1GB of memory, you should add the `/3GB` switch to the `BOOT.INI` file of your server. Windows NT and Windows 2000 allocate 2GB of virtual address space for the Windows NT kernel and 2GB of virtual address space to user mode processes. Windows 2000 Advanced Server and Windows 2000 Datacenter allow you to change this allocation with the `/3GB` switch in the `BOOT.INI`. This feature is not available with Windows NT 4 or Windows 2000 Server.

The Exchange information store is typically the memory hog on an Exchange server. If the server has 2GB of RAM, the store will attempt to use as much of that memory as it can. However, as it allocates RAM, the amount of virtual memory it has allocated will be greater than the actual amount of RAM. It will run out of virtual memory even though there is still RAM available. Memory allocations will begin to fail, and the information store service will have to be shut down in order to resolve the problem. However, the problem will recur when the information store has restarted. The solution is to make sure that Windows 2000

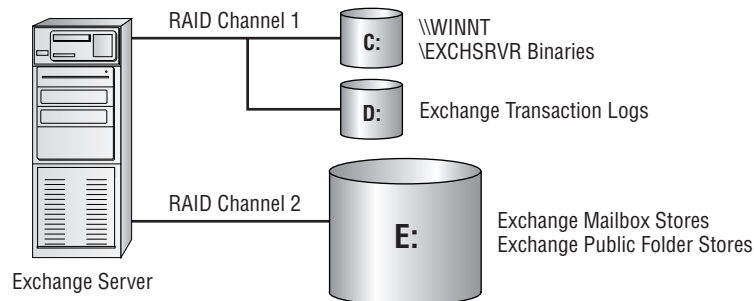
Advanced Server or Windows 2000 Datacenter has the /3GB switch in the BOOT.INI file. If you are not using Windows 2000 Advanced Server or Datacenter, you should upgrade the operating system.

**NOTE** For more information about the /3GB switch, see Microsoft Knowledge Base article Q266096.

### Locating Exchange Data and Transaction Log Files

One of the keys to improving Exchange performance is to ensure that Exchange database and transaction logs are located on separate physical hard drives, as shown in the simple server configuration in Figure 1.7. The operating system and Exchange binaries are located on one physical disk (hopefully mirrored), the Exchange transaction logs are located on a separate physical disk (also mirrored), and the Exchange public folder and mailboxes stores are located on a separate physical disk (in this case, a RAID 5 array).

**Figure 1.7** Disk drive configuration for Exchange



Significant performance improvements will always be realized by placing the transaction log files on a separate physical disk. The disk should not be shared by any other application. If you create more than one storage group, each storage group's transaction log files should be placed on a separate physical disk drive. The volume on which you place the transaction log files will best perform on a mirrored volume or a RAID 0+1 volume (mirrored, striped sets). During normal server operations, log file usage is exclusively write-intensive, so RAID 5 drive arrays will hurt transaction log file performance.

If the Exchange server functions as a bridgehead server, then the \EXCHSRVR\MTADData and \EXCHSRVR\Mailroot directories should be located on a separate physical hard disk. A single physical disk dedicated to the Mailroot and MTADData directories should be sufficient for all but the most extreme circumstances (thousands of message per hour to both the MTA and the SMTP message system).

## User Education

An area that is often overlooked during deployment of a new messaging system is end-user training. When you are a technical person managing the Exchange deployment, it is easy to rationalize that your end-user community can “figure things out.” This is a bad assumption. My advice is to plan for two phases of training.

The first run of training should address equivalent functionality issues. Show the users how to use Outlook and Exchange to get the identical functionality that they had previously. Your user community is going to have enough worries adapting to a new software product; teach them just what they need to know to do exactly what they were doing with the old system. Many companies also provide “floor support” for their users immediately after training. This is a trainer or person who is already comfortable with the new system. Throughout the workday, the floor support person checks with the users who have just started using the new system to see if they have questions or need assistance.

Once your user community has become accustomed to the new system and to the basic features of Outlook, the second phase of training introduces them to features they may not have had previously, including group scheduling, rules, journaling, and task assignment. Additional phases of training can introduce new features such as using forms and public folders.

### **Exchange@Work: A Training Plan**

What is one great way to reduce your help desk costs? Turn your end users into power users! Company GHI migrated a 2400-node network from Lotus cc:Mail to Exchange. Prior to migration, the cc:Mail users had no features such as calendaring or group scheduling available to them.

The first phase of user training introduced the basic abilities of Outlook. The three-hour mandatory training session covered:

- The acceptable use policy for using the system, keeping mailboxes cleaned up, password and confidentiality issues, and help desk procedures. Also discussed were the introductory training materials, which included a short “How to...” guide and frequently asked questions section.
- Message formatting, attachments, and other features of an Outlook message such as delivery and notification receipts.

**Exchange@Work: A Training Plan (continued)**

- How to send, receive, reply to, and forward e-mail messages using Outlook; how to manage the Inbox and Sent Items folders; and how to create subfolders to better organize messages. Searching features and views were also introduced.
- Contact and Calendar folders (with the promise to users of future calendar training).
- Exchange's public folders feature vs. cc:Mail bulletin boards. Three public folders were discussed: a system announcements public folder, a Microsoft Office tips and questions public folder, and a classified ads public folder. (This last folder was specifically designed to stimulate people's interest in public folders.)

Once all users had been migrated and trained on basic functionality, another two-hour mandatory training session was held for all users that included:

- Group scheduling and calendaring (as promised)
- Scheduling shared resources such as conference rooms, laptops, and so on
- Creating a personal folder (PST file) and archiving messages to it
- A vacation/time-off request form
- A new public folder application, departmental In/Out boards

After the system had been in production for nearly nine months and users had been given time to get very comfortable with the basic features of Outlook, weekly training sessions were offered. Each "no nonsense" session covered one particular topic in detail. The sessions were offered five times during a week (at lunchtime), and the users were encouraged to bring their own lunch (they were called Brown Bag sessions). Though this got off to a slow start, the Brown Bag sessions became immensely popular and were often standing-room only. These one-hour meetings included topics such as:

- The Rules Wizard and the Inbox Assistant
- The Outlook Journal feature
- Accessing other users' mailboxes (calendars, contacts, and so on) or giving other users access to a user's mailbox

**Exchange@Work: A Training Plan (continued)**

- Outlook usage for remote or home users
- Outlook refresher courses

Well, you get the idea. GHI had implemented this particular strategy when converting from WordPerfect to Word with excellent results, and they continue to offer Brown Bag sessions for Word, Excel, and Outlook. This is a great example of a company doing whatever it takes to turn their user community into power users. GHI realizes that in the long term, better-educated users will reduce the total amount of IS support they'll require.

## Messaging Champions

In any department or workgroup, a few users inevitably arise from the ashes of a new mail system as champions of the new technology. These folks see early on the benefits of Exchange and Outlook and become evangelists for your cause. You should identify these people early on and encourage them. Often, these users will end up being your “on-the-spot help desk.”

## Getting the Training Done

A lot of companies today have an in-house training staff. If this is the case with your organization, make sure that your in-house trainers are brought into the Exchange design process early. The trainers should have been using the server and client features long before they start training the user community.

If you decide to contract an outside organization, carefully select both the company and the individual(s) who will be providing your client training. The trainers should understand the client (such as Outlook) as well as Exchange Server. If possible, look for a training company that has experience with your legacy system and is amenable to customizing the training to suit your users' level of expertise and needs.

Once you have selected your training company, ask if you can work with the same trainers throughout your training process so you'll have an opportunity to familiarize them with your existing system and procedures. The more comfortable the third-party trainers are with your organization, the better training your users will receive.

## Other Training Topics

What should be covered in training other than using the messaging system? What can you do in end-user training that will help keep your Exchange system healthy? I am betting that there is a long list of things you wish your users knew, and many of those things are not technical. In addition to the obvious, some things that I would make sure to cover during training are

- Showing users how to use distribution lists (or to use the appropriate public folder rather than a distribution list).
- Instructing users to double-check their To, Cc, and Bcc fields to make sure that the message is addressed to the proper recipient(s).
- Teaching users the difference between the Reply button and the Reply To All button. Discourage the use of Reply To All. Users should ask themselves “Does everyone who originally received this message need to see my reply?” Inevitably, a few times a year, a user will hit the Reply To All function by accident and reply with a message like: “I’d like to make it to the company picnic, but I’m getting a wart removed.” For the accidental recipients of this message, this is too much information.
- Discouraging large message signatures and signatures with graphics in them.
- Showing users how to send shortcuts or links to large files rather than attaching the files directly to the message.
- Storing files in personal folders, archiving messages and other Outlook data to these files, and retrieving data from personal folders.
- Reviewing the acceptable use policy.
- Educating users about chain letters and urban legends so that they will not be so quick to forward “Good Times” virus warnings, pictures of naked celebrities, free money ads, and warnings of kidney theft rings.
- Teaching users to delete messages once they have acted on them and to keep their Sent Items folder clean.

## Documentation for End Users

Another integral part of the training process is providing your user community with detailed documentation. This guarantees that you don’t leave your users out in the cold during and after the migration. Their first line of defense will be your help desk, but you want to give your users something they can use *prior* to calling the help desk. This may be a simple handout or a complete manual. This documentation should include:

- “How to...” guides for common tasks
- Frequently asked questions (and answers)

- Common problems and how to resolve them
- Special notes on what you learned during the pilot project

For example, one small company that I worked with assembled this material in a very professional, bound booklet. The user was given this booklet a few days prior to attending training. I personally like hard copies of my reference material, but a web site or an Exchange public folder is also an excellent location for posting this material.

#### **Exchange@Work: Can You Give Me Just a Hint?**

A particularly resourceful network administrator created one-page handouts describing how to perform certain tasks in Outlook, Word, Excel, and so on. These handouts focused on a single hint or task, included a graphic or picture, and were never more than a single page.

She posted these in the employee kitchens, photocopy rooms, above the fax machines—any place people would stop for a few minutes and might have idle time. I suspect she even tried posting these in the elevators and bathrooms.

At first, the user community did not know quite what to think. However, her writing style, humor, the concise nature of each “page,” and the usefulness of the information proved quite effective. She recently created an internal web site of her “greatest hits”; unfortunately it is not accessible outside of her company.

### **And a Little Bit of Therapy on the Side...**

Migration to a new message system is a stressful experience. As system administrators and network engineers, we all recognize this type of stress. We have deadlines to meet, executives and managers demanding successful implementations in one or two days, and impatient users.

As technical people, we often overlook the fact that a migration is also stressful for the end users. We technophiles view upgrades as a way to get to work with new things. Every few months I have to listen to one of my friends complain about changes in their computer system at work. “Nothing works the way it used to.” “The IT department never tells us anything; they just show up and change things.” “Why do I have to learn something new?”

End users do not view system upgrades with the same optimism that we do. They view technology as the means to do their job, not the job itself. Once they learn to manipulate a tool to do something, even if it is a poor tool, they are often reluctant to upgrade to a better tool.

Some larger organizations have someone on staff who analyzes the changes in people's work environments. If this modification will introduce too much stress, this staff member works with the facilitators of the change to make sure that the negative impact is minimal.

You can help minimize stress on your user community by providing them with good documentation as well as by getting their input—and letting them know what is going to happen and when. Once you create a schedule, stick to it! Let the users know that you care and that you will do everything you can to make sure they are happy with the things that are about to change. If all else fails, the users are burning you in effigy, and things are going terribly wrong, I suggest a little Rocky Road ice cream, a chili dog, and some time with a surfboard.

