Realtime
publishers

"Leading the Conversation"

The Shortcut Guide™ To

Exchange Server 2007 Storage Systems

Jim McBee

Copyright Statement

Chapter 2: Sizing the Exchange Storage Platform Appropriately

To the uninitiated, planning for Exchange storage, choosing the right number of disks, and selecting the right storage platform seem like simple enough tasks—throw a few mirrored disks at a fast-enough server and you are done! Indeed, if you are planning the storage requirements for an Exchange 2007 server that will support 100 mailboxes, "throwing a few disks together" will more than likely accomplish your goals.

Why then do so many Exchange Server installations run out of disk capacity far before a company had planned? Why do users start complaining about bad performance after a few months of using Exchange? Why do some storage designs not mesh well with the rest of an organization's storage plans?

Planning for Exchange storage can be a bit tricky in a few regards. The first is planning for storage capacity. The bottom line is that users will almost always ask for more storage than you really want to give them. In some cases, they may actually need more storage in order to efficiently do their job.

The second Exchange storage quandary is that even when the amount of storage that is required for email is calculated accurately, there is actually more overhead associated with Exchange Server data once you consider factors such as deleted information retention, full text index files, database white space, and transaction logs.

One of the most common problems that Exchange Servers experience, though, is a lack of disk I/O capacity. For organizations with more than a few hundred mailboxes or for those that support heavy email users, there can be quite a heavy burden placed on the disk system. When the disk storage system is overburdened, users will begin to notice slowdowns in Exchange Server response and mail delivery times. Frequently, I/O capacity is not even considered during the hardware specification determination and design of an Exchange Server. This chapter will review not only the common mistakes that are made when calculating the amount of storage required to support a group of users but also the mistakes made when calculating the I/O capacity required to support that group of users.

Estimating Disk Storage Requirements

Let's first explore how to correctly estimate disk storage requirements and ensure that you don't run out of disk space in the first year of operation. This task involves calculating estimated storage, determining everything else that will require disk capacity, and adding your own "hedge factor." Before delving into disk storage requirements, consider the following email phenomena:

- Users, like nature, abhor a vacuum. They will allow their mailbox to expand and fill their environment.

- Sometimes users actually need all the email content they keep and sometimes they don't. If there is a legitimate business reason for granting someone more email storage, they should never be turned down.

- VIPs, managers, executives, and even some knowledge workers will have the political capital to have their mailbox limits increased over the default limit you have set for them. There is just no way around that. Plan for increased limits on some mailboxes.

- Email messages are larger than they used to be. Credit should be given to disclaimers, personal message signatures, S/MIME digital signatures, HTML and rich-text formatting, graphics, and, of course, attachments. The average message in my Inbox is around 55KB.

This chapter will use an example company to calculate the amount of storage (and later I/O capacity) necessary for a specific Exchange environment. These figures are based on a composite of a number of customers with which I have worked and some of Microsoft's official recommendations. This is the loading factor that I use when helping a customer estimate disk storage and I/O capacity. This fictional organization has 1000 mailboxes; 250 of these users are power users or VIPs. Email is an important application for this company and its users. Collectively, the company's users make heavy use of Exchange Server. Additional particulars include:

- During the peak of the business day, the concurrency factor is 1000. All 1000 mailboxes are being used. Although such is not always the case, this will be the consideration used to design for this concurrency factor.

- Of the 1000 users, 50 users carry a RIM BlackBerry device and 75 users carry a Windows Mobile device.

- The average message size is 50KB.

- The organization receives approximately 40,000 messages per day or about 40 messages per user.

When specifying your expected capacity, always lean in the direction of the maximum capacity that you expect your user community to generate. In my experience, as user communities mature, they start using more and more capabilities of an email system and thus use more of its capacity. Underestimating the capacity will result in servers running out of disk space, exceeding available CPU capacity, and exceeding disk performance capacity.

However, balance your expectations with your budget. Building a server that will support three, four, or five times the actual capacity you need results only in spending significantly more money than is necessary.

Common Mistakes When Sizing Storage

The first and perhaps biggest mistake that organizations make when estimating storage requirements is that they throw a lot of disk space at the server without performing actual calculations to verify that the disk subsystem will actually sustain the I/O load required by the users.

🖉 Later, this chapter will introduce the concept of proper I/O sizing.

Another common mistake is that the decision to allocate storage is made in a vacuum. Mailbox sizing decisions are made based on the IT department's desire to keep databases small, keep backup/restore times short, or meet a service level agreement (SLA). Sometimes a server or storage platform is chosen prior to making capacity decisions and the platform provides no room to grow. Mailbox storage capacity decisions must be made in concert with management, determining the types of data that must be stored and how long that data must be accessed.

A Common Goal

I once sat in on a meeting in which a senior manager asked the IT manager for a larger mailbox. She explained that she frequently accessed documents via Outlook and Outlook Web Access (OWA) and that 100MB mailbox limit meant she frequently had to delete things in order to keep her mailbox below the limit. This manager received request-for-proposal documents, contracts, and vendor information; I worked with this manager on a regular basis and knew she was not one to keep MP3 files, jokes, and funny pictures in her mailbox.

The IT manager's response was "it is easier for us (the IT department) if we keep the mailboxes small." The IT manager was completely missing the point of IT: to provide users with the information and services they need to do their jobs. Although IT professionals cannot simply give users whatever is requested, it is necessary to balance their needs with what IT can reasonably provide. If you don't have the capacity to do so, you need to examine how to expand your capacity to meet users' needs.

When planning Exchange Server storage capacity, take a hard look at exactly what your users will require and balance that with what you can realistically provide. If there is a big gap between what you can provide and what users actually require, you need to revisit your budget, server platform, and storage platform capabilities.

The following list highlights additional mistakes that are commonly made when planning for Exchange Server storage:

- Incorporating PST files into your design. PST files are not intended to be used as an enterprise storage or archival media.
- Overlooking capacity planning for transaction logs, the operating system (OS), Exchange binaries, or the Windows page file.
- Failing to consider how white space, deleted items, and deleted mailboxes affect the size of the database.
- Failing to consider the size of full text index files when planning disk space requirements.
- Placing transaction logs and database files on the same volume or LUN.
- Using a storage system that does not easily allow for expansion.
- Forgetting to allow for extra disk space that may be necessary if you need to perform offline maintenance or are using volume shadow copy backups.
- Failing to plan for disk space for message tracking and protocol logs.

OS Disk

One of my pet peeves is working on any Windows server that has fewer than 10GB of disk space configured for the OS disk (usually the C drive). After a few short months of operation, I usually find myself fighting to keep enough free disk space.

There are a number of pieces of software, components, or other files that you may put on the OS disk that will affect the available disk space. Although many of these can be moved or assigned elsewhere, often administrators don't think about doing so until they are short on disk space. The following list provides a sampling of software and components that you may find on the OS disk:

- The OS can easily exceed 3GB of disk space depending on the components installed.

- User profile directories are found in the \Documents and Settings folder. In an environment in which there are many administrators that user the Remote Desktop Connection, there may be a lot of user profile folders.

- The Exchange binaries default to the OS disk. Depending on the server roles installed, the Exchange binaries' disk space usage can exceed 1GB.

- The OS page file resides on the OS disk by default. The page file should be the amount of RAM plus 10MB. For a server with 16GB of RAM, the page file will be just over 16GB.

- The OS TEMP files reside in the C:\windows\temp folder. On a hub transport or edge transport server that supports a large number of inbound and outbound messages, this %TEMP% variable should be moved to another physical disk.

- Exchange message tracking logs, protocol logs, and the hub transport databases may also be located on the OS disk. On a server with less than 1000 mailboxes, it is usually okay to leave these on the same disk as the Exchange binaries and OS.

In addition, a useful practice is to keep a copy of the installable software that you might need to maintain a server, including a copy of the Exchange Server DVD (or CD-ROM in the case of Exchange 2000/2003), message hygiene (antivirus and anti-spam) software, the OS install files, tape backup installers, SAN drivers, iSCSI drivers, and anything else necessary for that particular server. This can easily be 10GB worth of software.

For servers on which the Exchange databases and transaction logs are on separate disks or logical units (LUNs) but the remainder of the files is on the OS volume, I recommend an OS disk size of at least 36GB. If you are supporting Exchange 2000/2003 servers and you use the Windows Backup Utility to back up Exchange to files (BKF files), plan to keep at least as much free disk space as the largest Exchange database that you will ever need to restore.

Page File Disk

On servers that support a large number of mailboxes (usually greater than 2000 simultaneous users), plan to put the page file on a different physical disk drive. Keep in mind that the page file should be at least the amount of RAM plus 10MB. This could be as much as 32GB just for the page file!

There is nothing magic about the 2000 user number. In some environments, 1000 users and server-side applications could generate a significant enough load to warrant moving the page file off the system disk; in other environments, 3000 mailboxes would not place a significant enough load on the OS disk to warrant moving the page file.

Transaction Log Disks

Each Exchange Server storage group has a unique set of transaction logs that serve the databases in just that one storage group. For Exchange 2003, the recommendation is each time a new mailbox database is required; create a new storage group until you have used up all four available storage groups. Only then should you go back and create additional databases in those storage groups. This is contrary to advice that was given by Microsoft for Exchange 2000, which was to fill each storage group with databases before creating a new storage group.

Exchange Server 2007 Standard Edition supports as many as five storage groups and five mailbox databases. Exchange Server 2007 Enterprise Edition supports as many as 50 storage groups for a maximum of 50 databases. Each storage group can have one database in it, so you could have 10 storage groups with five databases or 50 storage groups with one database each. Although you could create a storage group and put as many as five mailbox databases in each storage group, the best practice is to create a storage group and then create one database for that storage group. Thus, an Exchange Server 2007 system requiring 25 mailbox databases would have 25 storage groups and 25 separate sets of transaction logs.

Increasing the Checkpoint Depth Available to Databases

There is a performance reason for spreading databases across many storage groups that relates to the concept of the *checkpoint depth*. The checkpoint depth is the maximum amount of data that can be written to the transaction logs before the database engine must commit data to the database file. The larger the checkpoint depth, the less frequently the database engine must flush data from memory into the database file. This reduces the I/O requirements for the database file because writes can be made more efficiently.

The default Exchange 2000/2003/2007 checkpoint depth is 20MB. Thus, if there are four databases in a storage group, each database would have approximately 5MB worth of data in the logs before the data would have to be committed to the database. This would require more frequent database write operations for each database. Consider as an example the Exchange mailbox servers that Figure 2.1 shows; there is a single storage group that contains four mailbox databases. All four of these mailbox databases must share a common set of transaction logs. The more databases in a storage group, the smaller the checkpoint depth for each database and the more frequently writes must be performed to the database disks.
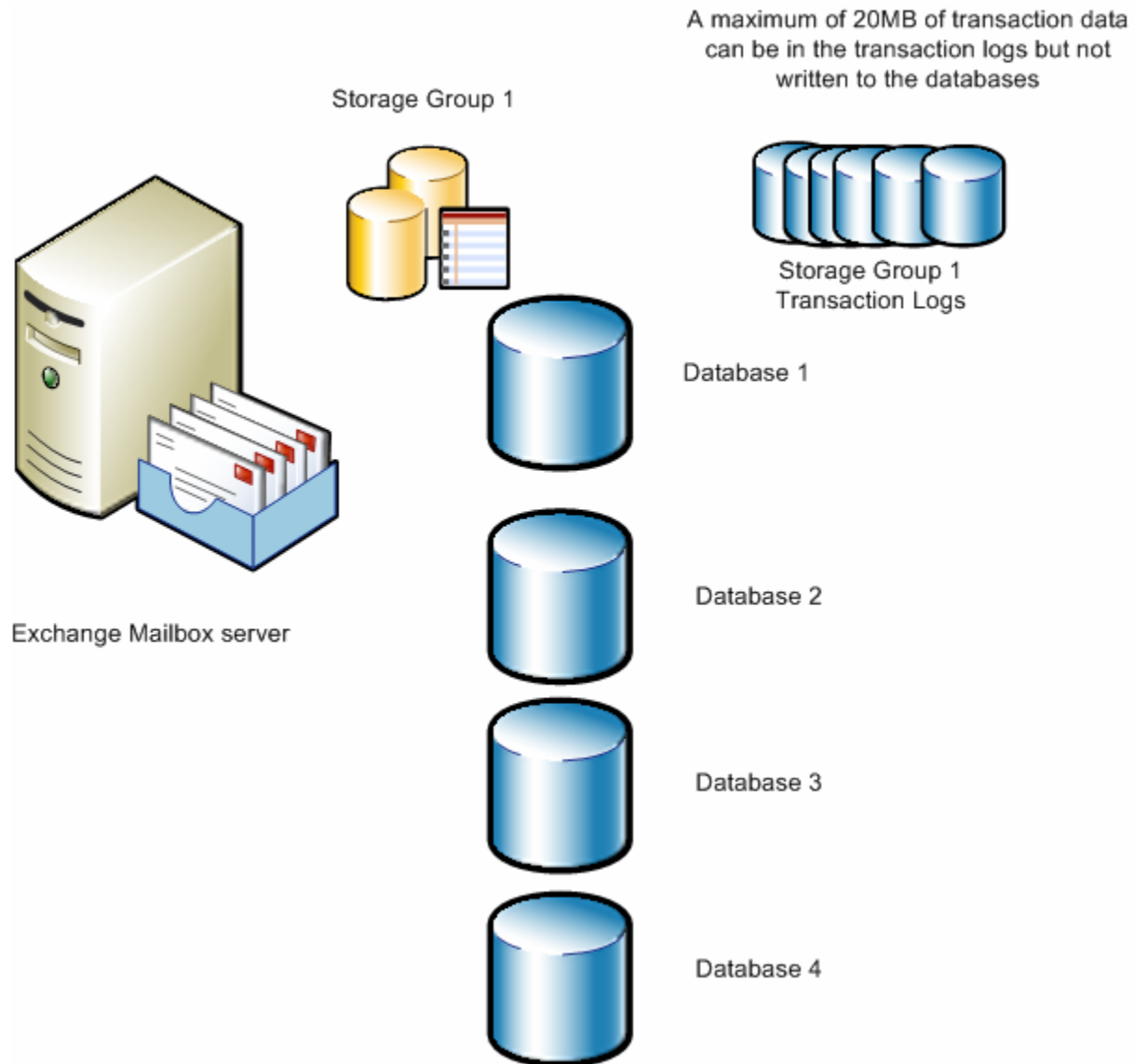
Figure 2.1: Multiple databases sharing a single set of transactions.

✎ Checkpoint depth for each storage group can be increased by using ADSI Edit. I recommend against doing so unless you have specific advice from Microsoft to make this change.

However, if you create four separate storage groups that each contains one database, the transaction log checkpoint depth for *each* database would be 20MB. Data can be stored in memory longer and written to the disk more efficiently. Figure 2.2 shows that each storage group contains one mailbox database; that one mailbox database can use the entire checkpoint depth for itself.
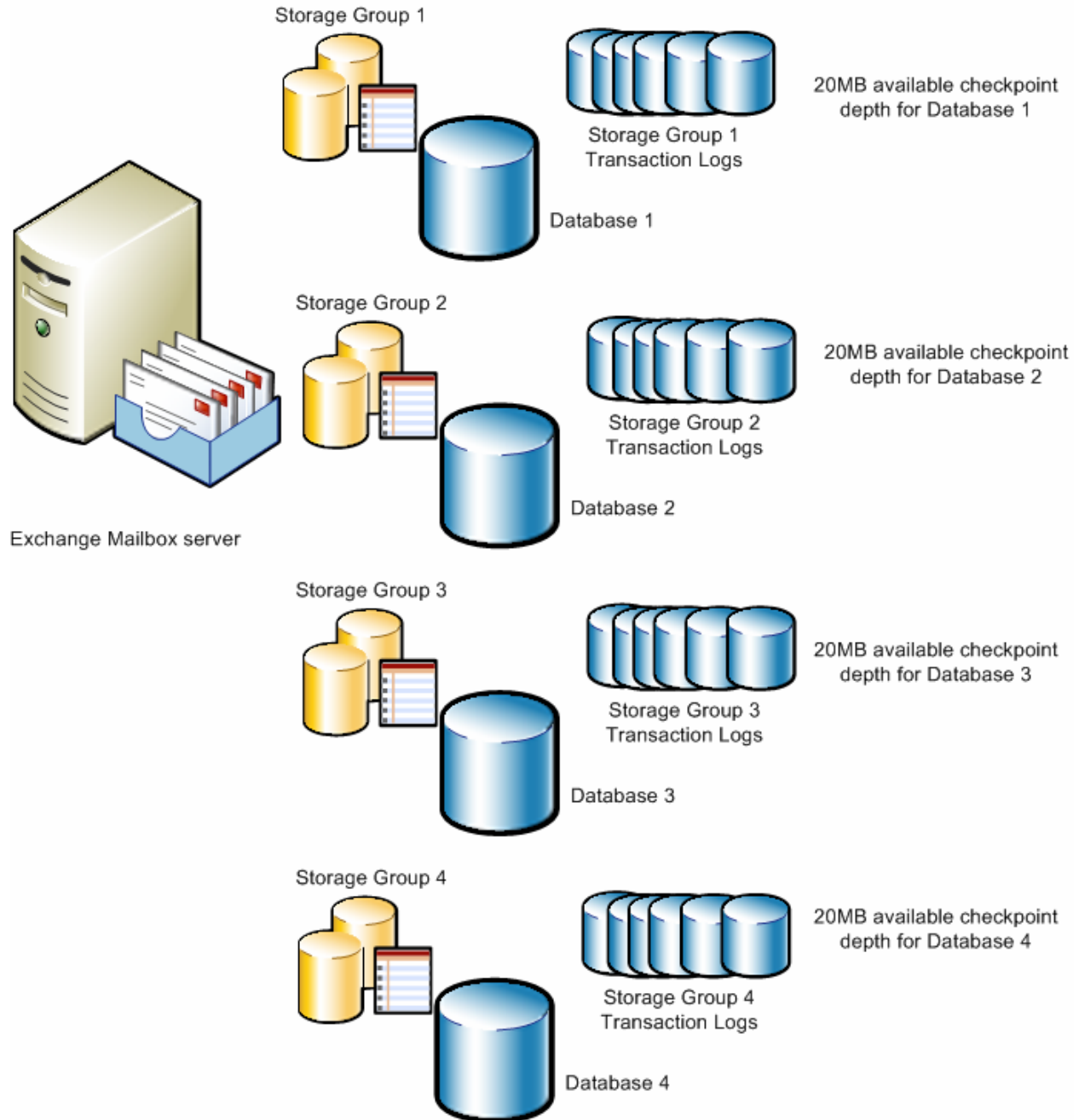
Figure 2.2: Putting a single mailbox database in a storage group increases the checkpoint depth available to that database.

By spreading databases across many storage groups, the average checkpoint depth available for each database is increased. Thus, cached transactions do not need to be flushed to the database as frequently. Disk writes can be made more efficiently and thus reduce the I/O requirements for the database disks.

Estimating Transaction Log Space Requirements

It is essential that you have enough transaction log disk space to allow the server to operate properly between normal backups. If circular logging is not enabled (it usually is not), the transaction logs accumulate until a normal or an incremental backup is performed. Once the backup has backed up the transaction log files, the Exchange extensible storage engine (ESE) will purge the old transaction logs.

For an Exchange Server with a medium-to-heavy user load, I typically plan for about 1GB of transaction logs for each 100 users or about 10GB of transaction logs per day for 1000 users. This is only an estimate; depending on your user community, you may see more transaction logs generated each day or less. Transaction log volume will also vary from one day to another depending on other factors such as daily mail volume and mailbox moves. I have seen organizations with 100 mailboxes generate 2GB of transaction logs on Monday and Tuesday but by Friday they are generating only about 500MB of transaction logs.

I also plan to hold at least 1 week of transaction logs on the server's disk. I do this in case there are backup problems and a normal or online backup is not performed. Transaction logs should not be deleted manually in most cases unless you are doing so under the guidance of Microsoft Customer Support Services (CSS formerly known as Product Support Services or PSS.) For the fictional company I am using as an example, I recommend a total amount of storage of transaction logs of at least 70 to 85GB.

Database Disk Space Requirements

Next let's start calculating the amount of disk space that you expect the mail database files and the full text index files to consume. Rather than calculating individual mailbox databases first, calculate the total amount of mail data storage you are going to require. This means that there are a number of factors you need to consider. You not only have to calculate the amount of disk space that the actual data will use but also other factors that affect the database fix size. This includes the maximum amount of storage you expect each user to consume, the amount of data in the deleted item cache, the amount of white space in the database, and the space consumed by the index files.

Defining the Mailbox Limits

First define the type of users you are going to have and the storage limits that you will set for each of these users. Defining storage limits is by far the most important thing that you will do in order to help keep your servers operating within an expected set of parameters; in this case, the expected parameter is maximum disk space. For simplicity's sake, the sample company has 1000 mailboxes, but 250 of these mailboxes belong to VIP users, power users, or users that for some reason (political or otherwise) require more space than a typical user. For a typical user, I will assign the following storage limits:

- Issue warning: 250MB or 256,000KB

- Prohibit send: 275MB or 281,600KB

- Prohibit send and receive: 350MB or 358,400KB

For the VIP and power users, I will give them quite a bit more mailbox storage space than I did the typical user; I will assign the following storage limits:

- Issue warning: 1.9GB or 1,991,680KB

- Prohibit send: 2GB or 2,097,152KB

- Prohibit send and receive: 2.3GB or 2,411,520KB

Some administrators may look at these VIP limits and think that they are fairly large. Indeed they are large mailboxes, but the changing nature of messaging may require such mailbox sizes for some users. For users that use Unified Messaging applications, forms routing, or other email-integrated workflow applications, this additional disk space may be absolutely necessary.

I have worked with a number of organizations over the years that resist putting in a send and receive limit. The send and receive limit closes the mailbox so that it will no longer accept mail from anyone inside or outside of the organization. I think this limit is important because it keeps mailboxes from continuing to fill up. You can set this limit significantly high enough so that just one or two messages won't take it from the prohibit send to the prohibit send and receive limit in a very short period of time.

Defining the limits early on in an Exchange deployment is important. In Exchange 2007, mailbox limits can be defined on the Limits property page of each mailbox database (see Figure 2.3).
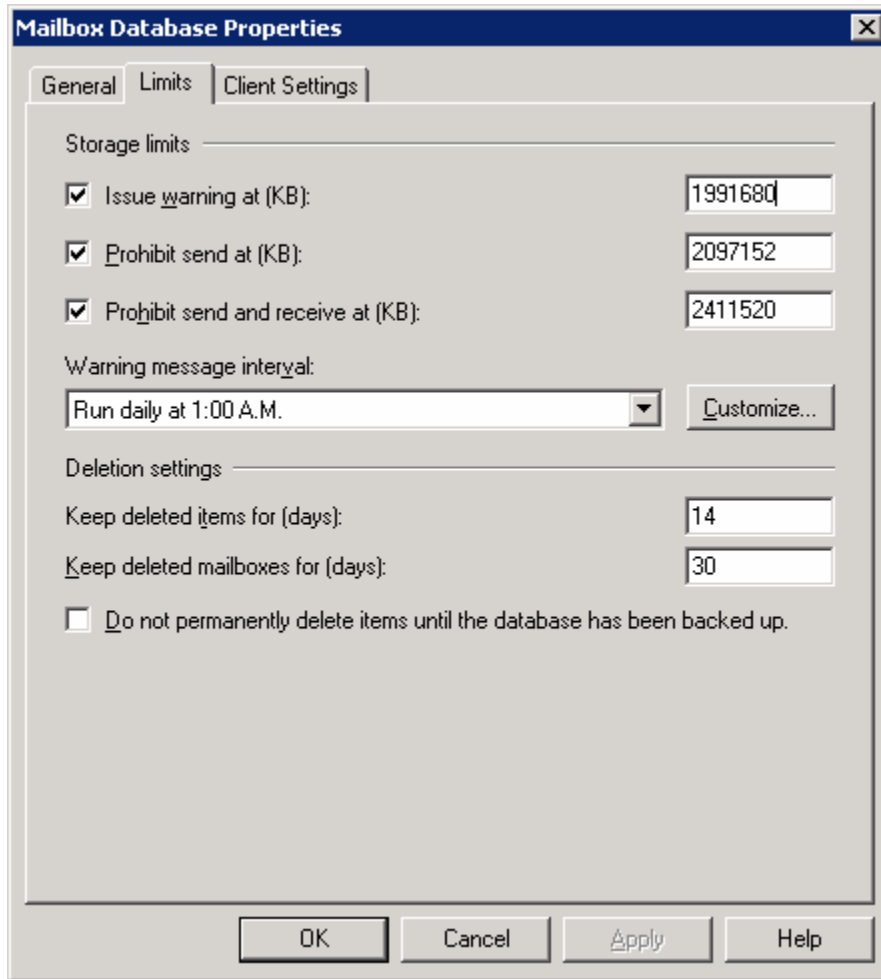
Figure 2.3: Assigning mailbox storage limits for an entire Exchange 2007 database.

Take a careful look at the storage limits; these are the default storage limits for a newly created Exchange 2007 mailbox database. No, you are not imagining things, the values are in kilobytes. The warning limit is 1,991,680KB (1,945MB or 1.9GB), the prohibit send limit is 2,097,152 (2,048B or 2GB), and the prohibit send and receive limit is 2,411,520 (2,355MB or 2.3GB!)

Individual mailbox limits can be set on each user's properties using the Exchange Management Console or the Exchange Management Shell. Figure 2.4 shows the Storage Quotas dialog box from the Exchange 2007 Management Console.
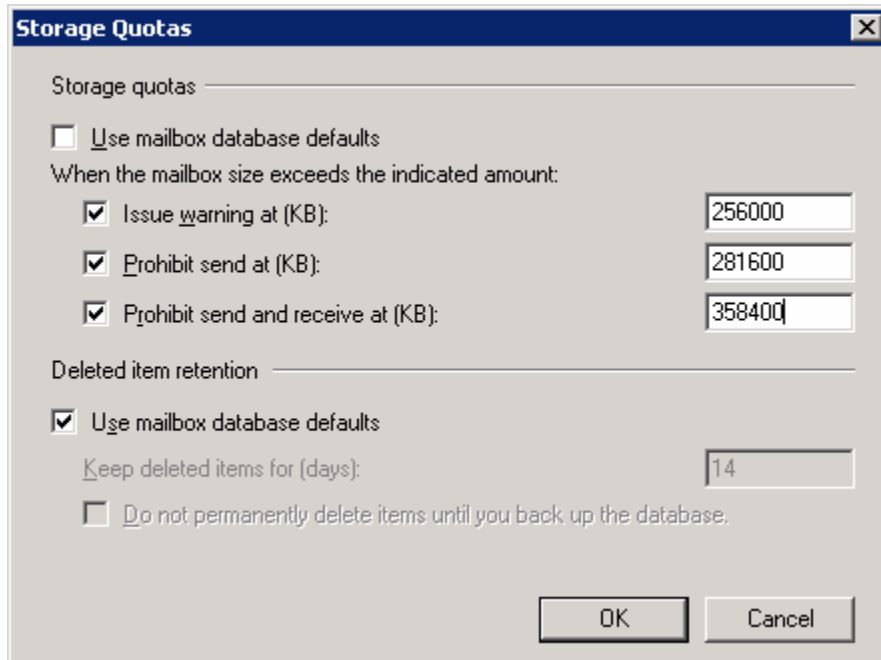
Figure 2.4: Individual user's storage quotas.

Limits can be defined in very similar ways for Exchange 2000 and 2003 using the Exchange System Manager console to manage the limits set on a mailbox store. Individual mailbox limits can be set on an individual mailbox using Active Directory Users and Computers.

In the following sections, I am going to attempt to predict how much space this user community will be consuming. This is a theoretical prediction based on experiences with Exchange 2000, 2003, and 2007 deployments and predicting "worst case" storage consumption; in other words, assuming that each user's mailbox is completely full. This is almost never the case for most organizations. In fact, many users that are given a few hundred MB of mail never get anywhere close to their limits. However there will be others that will be hovering near their maximum mailbox size from the day you create the mailbox.

Estimate the Maximum Amount of Data

Based on the established limits for regular and VIP users, you need to figure out the total, worst-case amount of storage that you will use. This is actually pretty because in this example there are only two sets of mailbox limits we need to worry about. First, we have 750 users that will have a maximum mailbox size of 350MB. This is a maximum total of approximately 263GB (350MB * 750).

For VIP users, the maximum mailbox size is 2.3GB and there are 250 of these users. This means that the maximum amount storage consumed by the VIPs of 575GB (2.3GB * 250). The grand total of VIPs and regular users is 838GB (575GB + 263GB) of actual mailbox content.

If you are looking at this number a bit skeptically, that is okay. It is a theoretical, worst-case maximum. I have never encountered a situation where every user's mailbox was at its absolute maximum size. Remember, though that we are designing for a worst case scenario. Further, we are not yet worrying about breaking up the mailboxes across multiple mailbox databases yet. I will get to that soon.

Estimate Full Text Index Usage

If you are taking advantage of Exchange Server's built-in full text indexing features, you also need to include disk space for the full text index files. Exchange 2000/2003 required that you enable each mailbox store to have a full text index; full text indexing is not enabled by default for Exchange 2000/2003. The size of the full text index files was between 20 and 40 percent of the size of the database; the number I use for estimating is 35%. For our sample company that has 838GB of mailbox data, the Exchange 2000/2003 full text index files (if all stores were enabled for full text indexing) would consume approximately 294MB of disk space. In my experience, full text indexing in Exchange 2000/2003 is not commonly used. This is due not only to the additional disk space required but also to the processor and disk I/O load required to keep the index up to date.

The full text indexing system in Exchange Server 2007 has been completely redesigned. Maintaining the index is much more efficient and disk space usage is significantly reduced. Full text indexing for each Exchange Server 2007 database is enabled by default. Microsoft says that the full text index files for Exchange 2007 consume approximately five percent of the size of the mailbox database; this has been consistent with my own experience with Exchange 2007. For the sample organization with 838GB of mailbox database, the full text index files will consume approximately 42GB of disk space.

Estimate Deleted Item Space Usage

Once mailbox items are deleted from a mailbox (either by emptying the Deleted Items folder or by performing a hard delete), they are moved to the deleted item or deleted mailbox cache. This feature is nicknamed the database dumpster. Data is retained in this dumpster for a length of time specified on the mailbox database's Limits property page (shown previously in Figure 2.3) or on the user's Storage Quotas properties (shown previously in Figure 2.4). Once this time has passed, the deleted item or mailbox is marked as permanently deleted and the space is recovered. This occurs nightly during online maintenance.

> ☞ Data in the database dumpster does not count against a user's storage quotas. However, data that has been recently moved into the dumpster may not be reflected in the user's mailbox size for an hour or two.

There are a number of factors that contribute to how much space will be in the dumpster:

- How long users want to retain the information they receive
- Mailbox sizes
- Message delivery rate and average messages size

If you are interested in measuring how much actual deleted space there is in an existing database, there are a couple of tools for doing this. For example, to see the number or the size of deleted items in the mailbox database, use the Windows Performance Monitor console and add the MSExchangeIS Mailbox object's Total Size of Recoverable Items counter. For this particular mailbox database shown in Figure 2.5, the actual database size is about 4GB so the amount of deleted space is almost two percent of the total size of the database.
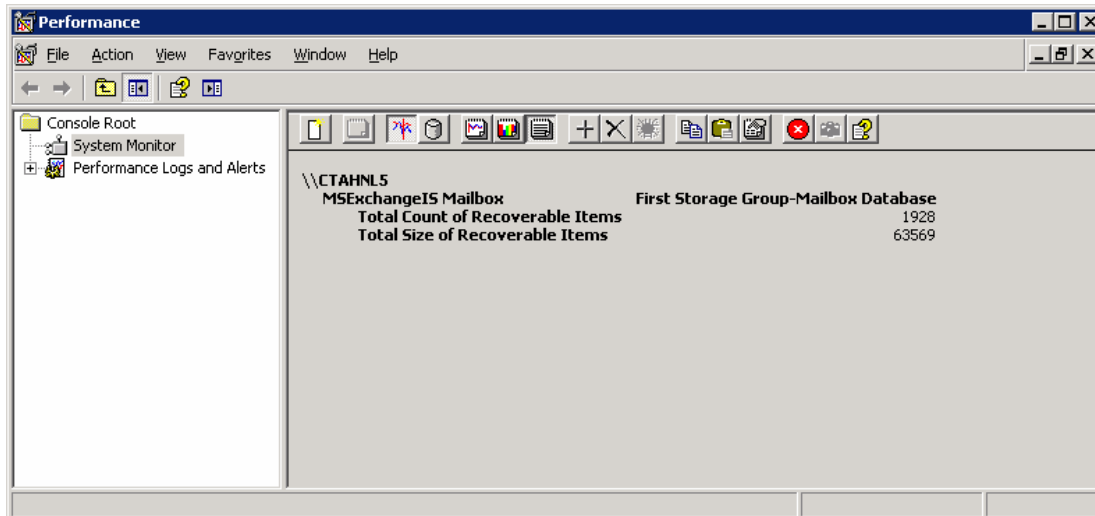
Figure 2.5: Viewing the amount of space used by deleted items.

You can see the same information if you search through the Application event logs for event ID 1207 from the MSExchangeIS Mailbox source. This event is generated during online maintenance. In Figure 2.6 you can see that for the database Mailbox Database there is approximately 63MB of deleted items.
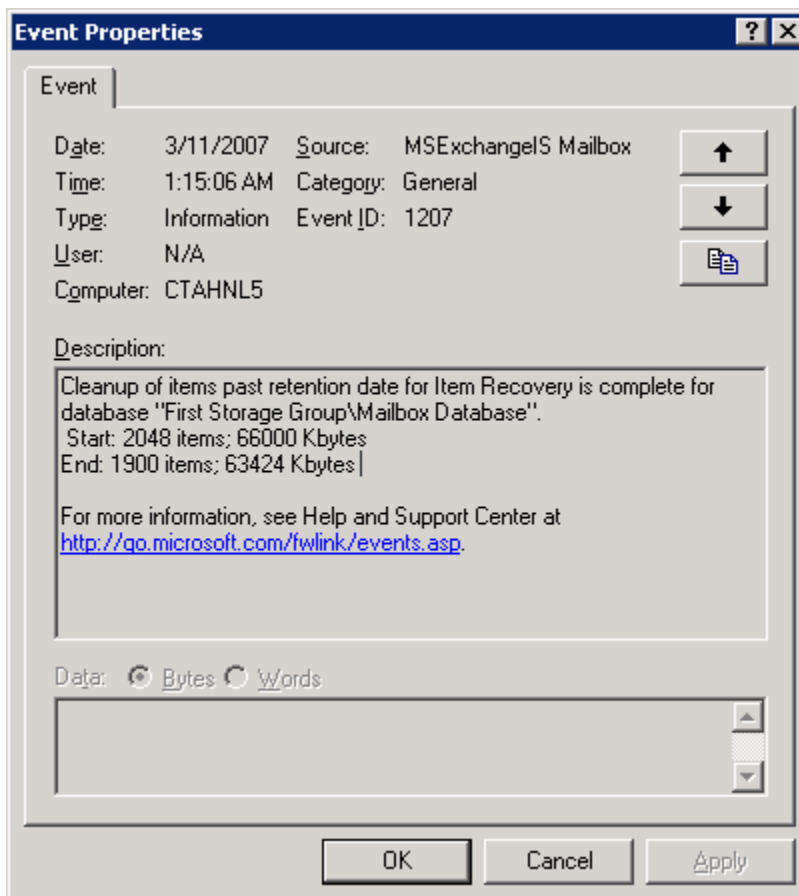
Figure 2.6: Viewing deleted item space through the event log.

Online maintenance will also generate event ID 9535 whose source is MSExchangeIS Mailbox. This event will tell you how much space is currently consumed by deleted mailboxes. Figure 2.7 shows this event. In this example, there are 26 mailboxes waiting to be deleted and they are consuming approximately 507MB of disk space in the Mailbox Database. In this case, this is approximately 10 percent of the total size of the database.
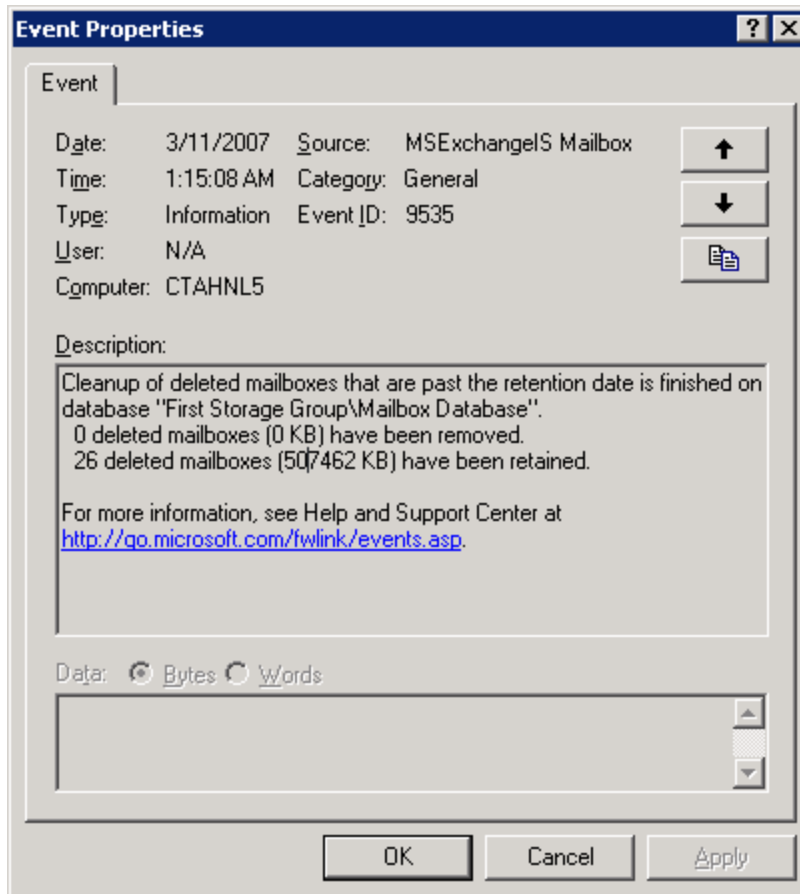
Figure 2.7: Viewing deleted mailbox space through the event log.

In Exchange 2003, deleted mailbox content is retained for 7 days and deleted mailboxes are retained for 30 days by default. In Exchange 2007, the deleted mailbox content is retained for 14 days and deleted mailboxes are retained for 30 days by default.

Coming up with a formula that will help you to estimate how much deleted item space is in your mailbox databases is difficult because this value will vary significantly over time. You may have to periodically remind your users to empty their Deleted Items folders or clean up old and unwanted messages. You may run the Exchange 2003 Mailbox Manager or the Exchange 2007 Messaging Records Management tools against mailboxes and clean up older content yourself. In academic environments, you might delete former students' mailboxes at the end of the school year and free up a lot of disk space.

One recommendation I have seen from Microsoft regarding estimating deleted item space is that the total dumpster size will be equal to the total size of incoming messages times the length of the deleted item cache. The total overhead, of course, would be factored against the mailbox sizes.

Here is an example using this formula. Let's say a typical user at an organization receives an average of 40 messages per day and the average message size is 50KB. Let's say this user deletes most of this incoming mail, but retains a few of these messages. Plus the user deletes a few older messages. The total deleted item cache for this user would be 2MB per day or 28MB at the default 14-day time period. For a user with a maximum mailbox size of 350MB, this would be approximately an eight percent overhead. For larger mailboxes, the percentage of overhead is even less.

Of course, most users are not so diligent about deleting their mail as they read it and no longer need it. Users usually delete their mail after they are nagged about it by an administrator or when they get a warning indicating that they are approaching their limit. However, in a mature messaging environment, this behavior may average out anyway.

I typically recommend using 20 percent of the total allowable mailbox sizes as the maximum amount of space. This allows for some 'wiggle' room and it estimates space used by deleted mailboxes for a typical organization. Thus, for the sample organization with 838GB of mailbox storage, there would be an additional 168GB.

Estimate White Space Usage

After deleted items and deleted mailboxes are removed from the database, the pages in the database are reclaimed during online maintenance. These pages are empty but are still part of the database file; the database file never shrinks after online maintenance completes. The next time the database engine needs free space in the database file, it uses the white space rather than growing the database file. You can see how much space in the database is white space by looking for event ID 1221 from source MSExchangeIS Mailbox. Figure 2.8 shows an example of this event.
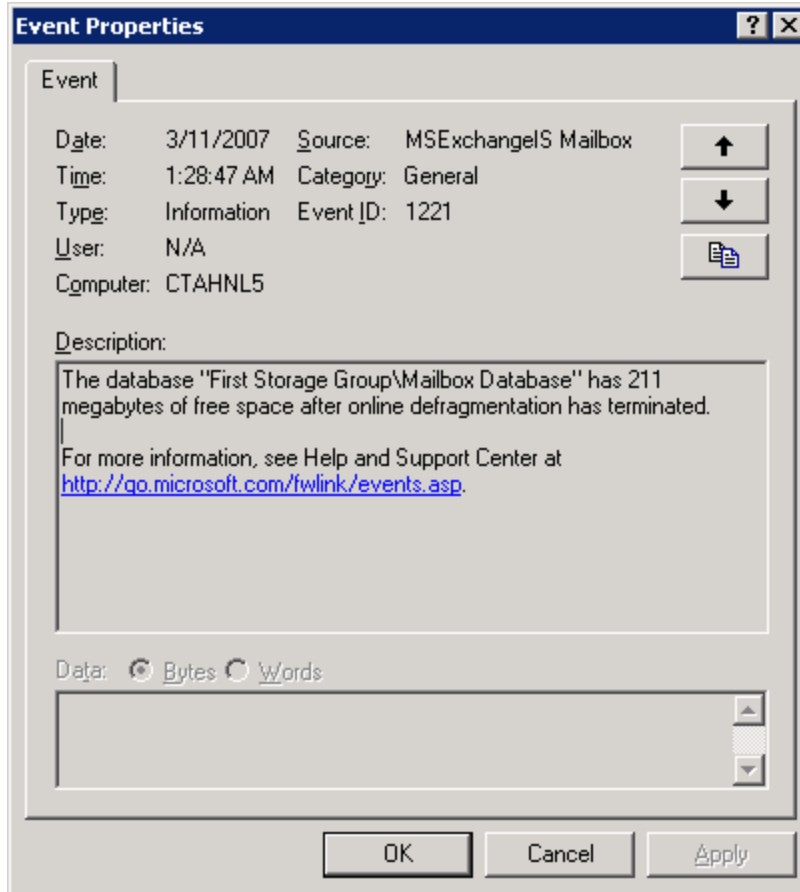
Figure 2.8: Viewing database white space.

In a perfect world, the whitespace freed up each day by online maintenance would equal the daily deletion rate for mailbox content. One estimate I have seen from Microsoft for use when factoring in database white space is that it will equal the daily messaging rate. Thus, in my example of a user community of 1000 that receives an average of 40 messages per day with an average size of 50KB, the daily size of message arrivals is approximately 2GB per day ((40 * 50KB) * 1000). As those messages are deleted, they are in the database dumpster for 14 days (the default period), and then online maintenance frees up that space. So on average, the database would contain 2GB of white space or in the case of a maximum database size of 838GB, the whitespace overhead would be less than .2 percent.

Unfortunately, just like predicting the database dumpster size, predicting white space is not always accurate. I have seen databases that vary between 1 and 20 percent of the total database size. This is often due to factors such as the fact that mailboxes are deleted in batches and message archival systems often run only a few times per month. This can result in the whitespace size varying quite a bit from one week to another.

I recommend using a 15 percent estimation factor when figuring out how much white space the database may consume. In the case of a database size of 838GB, there may be an additional 126GB of disk space consumed by white space.

Data Overhead

Wouldn't it be nice if users would live within the boundaries that you set for them? In every environment I have worked in, a few users end up getting more mailbox space than was originally planned. In the previous calculations, we have estimated the absolute worst-case disk space usage for mailbox users. However, if you start granting a few exceptions, it could actually get worse. That is why I like to include an additional overhead factor in the disk space usage calculations. I find that 20 percent is a good factor for estimating unexpected growth.

Adding it All Together

Based on the factors that affect database size, the following factors will affect the amount of Exchange data that will need to be stored in an absolute worst-case scenario:

- Total mailbox data is 838GB

- Recoverable item and mailbox space is 168GB

- Database white space is 126GB

- Data overhead factor of 20 percent is 168GB

Thus, the actual size of an Exchange database (if you were to put everything in a single database) is 1300GB or 1.3TB. Further, you need to plan for an additional 42GB for the Exchange Server 2007 full text index files.

Estimating Database Sizes

I have arrived at a worst-case, maximum amount of mail storage required of 1300GB of mailbox database capacity and 42GB of full text index capacity. You probably don't want to plan for a single 1300GB database, though; instead, split the storage across different databases. I have never seen an official recommendation from Microsoft on the maximum size of an Exchange 2000/2003 database, but sizes between 35GB and 50GB are frequently tossed around by respected Exchange gurus.

> ☞ Each organization's maximum database size may vary. A prudent Exchange administrator bases the maximum size on a value that can be backed up and restored in an acceptable amount of time. Remember that restore times may be twice as long as the backup time for a database when using streaming restores from tape. The acceptable amount of time will vary from one organization to another depending on SLAs and the restoration of mailboxes that are crucial to business operations.

Microsoft is recommending a maximum database size of 100GB if you are not using local continuous replication (LCR) and a maximum size of 200GB if you are using LCR. The concern is not for the stability of the database because an Exchange database can grow quite large and still be stable, but restoration times. Using LCR enables you to restore a database from the LCR copy very quickly.

For Exchange 2000/2003, I use 50GB as a maximum database size, and for Exchange 2007, I use 100GB as a maximum database size. However, still factor in the restoration time of the database. I calculate how long it would take to restore a database from backup media and determine what the SLA requires for a database restoration. In my humble opinion, a good "maximum restore time" value to use is 1 hour. If that requires databases smaller than 50GB or 100GB (depending on the Exchange version), so be it.

☞ If you are using streaming backups to tape, database restore times may take up to twice as long as the backup.

In this example, I will assume that the backup system can restore a database in under 1 hour, so I will estimate a maximum database size of 100GB. Since the maximum amount of data, white space, and dumpster space we estimated is 1300GB, this is going to give us 13 separate mailbox databases.

Now, you need to decide how you are going to spread the user community across these databases. In the Exchange 2000/2003 days, I recommended creating multiple mailbox stores for regular users and mailbox stores for VIP users. The maximum mailbox sizes were controlled by the Limits property pages on each mailbox database or by creating Exchange 2000/2003 mailbox system policies.

In some organizations, specific departments or groups of users are assigned to specific mailbox stores; however, this does not seem to work much better than just randomly and evenly distributing mailboxes across all mailbox databases. For Exchange 2007, I recommend creating all the mailbox databases necessary and assigning the regular user storage limits to those databases on the Limits property page. This can also be performed using the Exchange Management Shell (EMS). Here is an example using the Set-MailboxDatabase cmdlet to set all of the mailbox databases on server HNLMB01 to the appropriate storage limits. You can use the Get-MailboxDatabase cmdlet to retrieve a list of all mailbox databases and then pipe that to the Set-MailboxDatabase cmdlet:

```
Get-Mailbox | Set-MailboxDatabase -IssueWarningQuota:256000KB
-ProhibitSendQuota: 281600KB -ProhibitSendReceiveQuota:358400KB
```

This command used KB (kilobytes) for the mailbox storage, but you could achieve the exact same thing with the following command:

```
Get-Mailbox | Set-MailboxDatabase -IssueWarningQuota:250MB
-ProhibitSendQuota: 275MB -ProhibitSendReceiveQuota:350MB
```

Once you have assigned the regular user storage limits to all mailbox databases, you can override the storage limits for the VIP users. Although you could certainly do this using the Exchange Management Console and edit each user's account, it is much easier to do this with an Exchange Management Shell command. In this example, I have created a group called VIP Mailbox Storage. I will enumerate the membership of that group using the Get-DistributionGroupMember EMS cmdlet and pipe it to the Set-Mailbox cmdlet:

```
Get-DistributionGroupMember "VIP Mailbox Storage" | Set-Mailbox -
UseDatabaseQuotaDefaults:$False -IssueWarningQuota:1945MB
-ProhibitSendQuota:2GB -ProhibitSendReceiveQuota:2355MB
```

The Exchange Management Shell can greatly simplify administration and consolidate tasks that would have once been many lines of code or several hours of work for a single command-line task. There is an initial intimidation factor for the Exchange Management Shell, but once you get over that, it is actually quite simple to use.

Estimating Disk or LUN Sizes

Finally, you are to the point where you have to estimate the amount of disk space necessary for the transaction log volumes and the database volumes. Remember that when you are calculating disk volume requirements, always put the transaction logs on a separate volume or LUN from the databases. This is both for recoverability reasons as well as potentially improving performance by isolating sequential write-intensive operations that are required for transaction log writes from the random read/write operations involved in maintaining the database. If you are sizing a server that will have local disks, you only need to assign local disk volumes. However, if the server will be using disks from the storage area network (SAN), you will need to assign LUNs for each volume required.

The example server is going to require a minimum of 13 mailbox databases. For Exchange 2007, I recommend using 13 separate storage groups; for Exchange 2000/2003, this will require all four storage groups. When you calculate the number of LUNs required on the SAN, one of the decisions you will need to make is whether you are going to do online streaming backups of the databases or take advantage of the SAN's snapshot capabilities.

If you are only going to be using streaming backups, you can group the databases onto different LUNs based on the backup schedule. For example, if you are going to backup mailbox databases one through four on Monday, databases five through eight on Tuesday, etc… then you would group the databases so that all databases in a particular backup set are on the same volume. In the case of streaming backups on Exchange 2007, you would need at least four LUNs because you would be placing four databases on each LUN.

However, if you are backing up the databases using a volume shadow copy service (VSS) backup technology, you need to consider the granularity of restoration. VSS backups and restores are LUN based. It is considered a best practice to assign each Exchange database to a separate LUN. In this case, you would need at least 13 LUNs for databases.

In addition, you need separate LUNs for each storage group's transaction logs. In this case, you would need 13 separate LUNs for the transaction log files.

This does not necessarily require you to have separate drive letters. To reduce the number of drive letters required, you should use volume mount points rather than unique drive letters. Actually, in this example, you are already at a point where you are requiring 26 drive letters and you only have 24 available. Look at the example shown in Figure 2.9. In this example, I am using only three drive letters (M: (for mailboxes), T: (for transaction logs), and L: (for other types of logs). The actual example with 13 drive databases and storage groups would require quite a few more LUNs, but you can see how this would scale.
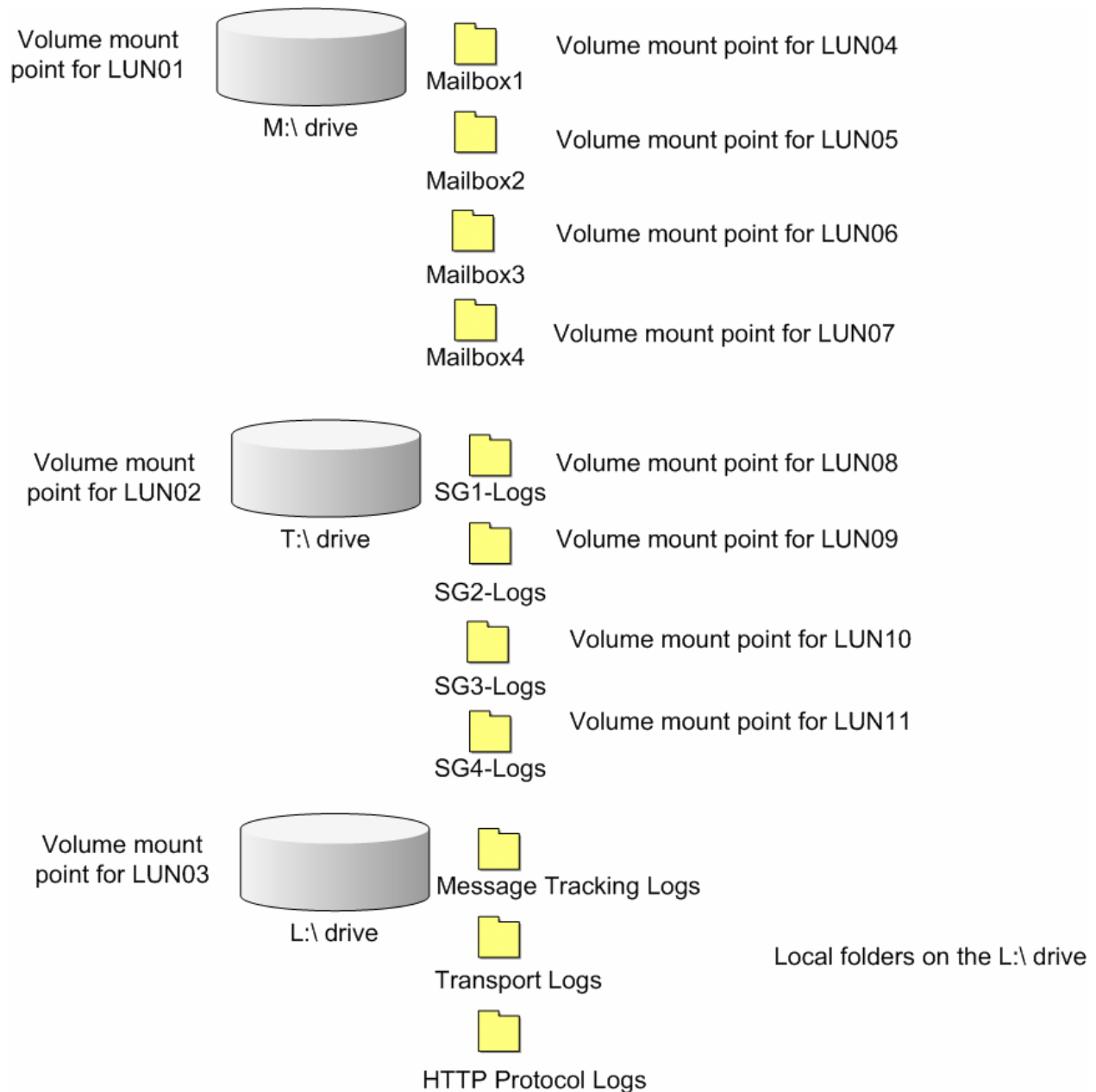

Figure 2.9: Sample drive letters and mount points.

Notice that the L drive in Figure 2.9 holds folders for the message tracking, transport, and HTTP protocol logs. I included these merely for illustration's sake. In the Exchange Server 2007 example, you would usually not include the Client Access server and the hub transport server roles on a Mailbox server with 1000 mailboxes. Thus, these log file folders would not be necessary.

However, on smaller servers that hold multiple roles or on Exchange 2003 servers, I recommend moving the queue files, hub transport databases, transport logs, message tracking logs, and HTTP logs to a separate LUN. These can all be separate folders on the same drive letter, though.

Each of these LUNs would need to be at least as large as the maximum size of the database that the LUN will contain plus the estimated full text index files. In this case, the maximum database size is 100GB plus another 5GB for the full text index files. I recommend creating the LUN size of at least 125GB. There is a conventional wisdom in Exchange storage design that you create a volume that will be at least 110% larger than the largest database size you expect to support. This allows for offline maintenance to be run and create the temporary files on the same LUN as the database is located. If you allow for this space on each LUN, the LUN will need to have approximately 220GB of disk space available. If you do not create this additional space, you will need to have some storage somewhere (preferably on the local server) that will be used for the temporary files that are created during the running of offline maintenance programs such as ESEUTIL or ISINTEG.

One of the advantages of choosing a storage technology that allows for dynamic growth of a LUN from available storage based on a set of constraints is that you can define the initial size of the LUN as reasonably small (in this case maybe 50GB). Keep a pool of several hundred gigabytes of available storage on the SAN. As a LUN requires more storage, it can automatically grow from that pool for storage. That way, if a LUN is required that is larger than you originally planned, it can grow as needed. Some SAN vendors support a feature called "thin provisioning" in which the SAN can automatically expand the LUN as needed without administrative intervention.

Other Factors that Affect Storage Capacity

To this point, the focus has mainly been on the factors that affect storage of mailbox data and the mailbox database transaction logs. This makes sense to focus on first because the mailbox data is by far the largest consumer of disk space; but there are a number of other factors and components that may increase your disk space requirements beyond a simple installation of the Exchange Server software.

Additional Factors Affecting Exchange Server 2003 Disk Space Usage

There are a number of factors that can increase the amount of storage required on an Exchange Server 2003 server. Many of these will depend on whether a particular feature is enabled or the role that a particular server is supporting. These factors include:

- Message tracking logs (disabled by default)
- SMTP logs (disabled by default)
- HTTP logs (enabled by default)
- SMTP queue folder

Additional Factors Affecting Exchange Server 2007 Disk Space Usage

The factors that affect an Exchange Server 2007 server are often slightly different that the factors affecting Exchange 2003. Part of this is due to the ability to segment server roles on to separate pieces of hardware. For example, in Exchange Server 2003, the message tracking logs and the SMTP queue folder exist on all Exchange Servers, but in Exchange Server 2007, message tracking logs are found only on servers supporting the hub transport server role. The following list highlights the different components that may affect disk space consumption and the server roles on which these factors should be considered:

- SMTP queue database on hub transport and edge transport servers

- Message tracking logs on hub transport and edge transport servers

- Transport and connectivity logs (formerly SMTP logs) on hub transport and edge transport servers

- HTTP logs on client access servers

- Sender reputation and IP filter databases on edge transport servers (or hub transport servers with the anti-spam agents installed)

- The unified messaging message queue on unified messaging servers

- Custom voice prompts and grammar files on unified messaging servers

Message Transport Databases

Exchange Server 2007 has a couple of new databases you will find on the hub transport or edge transport server roles. These databases include the queue database, the IP filter database, and the sender reputation database. The IP filter database and the sender reputation database will be relatively small (usually less than a few hundred megabytes); however, the queue database, which replaces the file-oriented queues used in Exchange 2000/2003, can grow fairly large. The IP filter and sender reputation databases are only found on the edge transport server or hub transport servers that have the anti-spam agents installed.

There are a couple of reasons that the queue database can grow significantly large. First, on both hub transport and edge transport servers, this database must hold mail that is not currently able to be delivered. This might be for as much as two days or even longer depending on how long you have configured the non-delivery interval.

The second reason that the queue database can grow is if you are using cluster continuous replication (CCR). The transport dumpster is used so that any loss that occurs as a result of a CCR failover can be replayed from the hub transport servers.

💣 Microsoft recommends that the disk that contains the message queue database always contain at least 4GB of free disk space. Otherwise, the transport system starts to apply "back pressure" and the hub transport or edge transport server will start to reject mail. Once back pressure is applied, the hub transport or edge transport server will start to reject mail.

Edge Transport Queue Databases

Factoring in the estimated size of the edge transport queue database is fairly complex. There are a lot of pieces that you need to consider including how many edge transport servers are running, the inbound and outbound delivery rate, and the NDR timeout (the period of time a message is held before it is returned to the sender as undeliverable—usually 2 days).

To keep this example as simple as possible, let's assume we have only a single edge transport server. In the case of the sample Exchange organization that we have been discussing, here are some important factors:

- The edge transport server receives approximately 40,000 messages per day or about .5 messages per second

- The edge transport server sends approximately 15,000 messages per day to the Internet or about .2 messages per second

- The average message size is 50KB

- The edge transport server will attempt delivery of messages in the queue database for two days before issuing a non-delivery report

If you assume the absolute worst case in which the edge transport is not able to deliver any outbound messages for 48 hours, but the internal hub transport servers continued to deliver email at the same rate, the edge transport queue database would contain approximately 1.5GB ((15,000*2)*50KB) of outbound mail.

If you assume that inbound mail continues to flow but the internal hub transport servers will not accept the inbound mail from the edge transport server, the edge transport server's queue database would contain approximately 4.0GB ((40,000*2)*50KB)) of inbound mail.

This mail flow rate is for a theoretical 1000 users using a single edge transport server for both inbound and outbound mail. These figures for the amount of mail that *might* queue up for inbound and outbound mail can pretty easily be scaled upwards. The theoretical worst case would be if neither inbound nor outbound mail would flow and the database would theoretically hold 6.5GB of data. I say this is the theoretical worst case because I can't really imagine a scenario in which mail would be delivered to the edge transport from the hub transport servers, but it would not be accepted by the hub transport servers.

To add some headroom or "fluff" factor to this calculation, let's just round up and say that for each 1000 users that an edge transport server will support, plan for a worst-case database size of 8GB (planning for a little unexpected overhead).

Hub Transport Queue Databases

The hub transport server has some of the same issues that the edge transport servers have with respect to receiving email from the Internet or delivering email to Mailbox servers or other internal hub transport servers. If an internal hub transport server or Mailbox server is down, the hub transport server will need to queue that mail for up to two days.

So let's start with a minimum of 6.5GB of data per 1000 users as the worst-case database size when supporting regular Mailbox servers. For the most part, this number is a little larger than practical, but it will give you some headroom.

This would be enough if you were only using regular Mailbox servers; however, if you are using CCR, you need to factor in the additional space consumed by the transport dumpster. After a message is delivered to the CCR clustered Mailbox server, the message is held in the transport dumpster for a certain period of time or capacity. The message is held until the transport dumpster reaches 18MB (for each storage group) or 7 days, whichever comes first. For each storage group that you are supporting, you must plan for approximately 18MB of mail. In the example we have been using, these 1000 users' mailboxes have been spread across 13 storage groups, so we will add an additional 234MB (13*18MB) for the transport dumpster.

Estimating Protocol and Tracking Log Sizes

The amount of traffic that passes through your Exchange 2003 servers or the Exchange Server 2007 hub transport and edge transport roles affect the growth and the size different log files that are created.

> ☞ Message tracking is enabled by default on Exchange 2007 servers; the logs are kept for a maximum of 30 days or up to 250MB, whichever comes first. The configuration is viewed using the Get-TransportServer cmdlet.

I always strongly recommend not only to have message tracking enabled but also to create protocol and connectivity logs. In Exchange 2003, the SMTP logs are enabled for each SMTP virtual server. For Exchange 2007, the connectivity and transport logs are enabled on the hub transport servers or using the Set-TransportServer cmdlet. My philosophy on logging is that you don't know that you are going to need them until an event you need to review in the logs has already passed. Build in the disk and system capacity to support the necessary logs!

To help you in planning for the amount of space that these logs might consume, I'm going to fall back on the example of the typical 1000 user organization that I have been using. Let's look at the size of logs they create each day:

- Approximately 25MB of HTTP logs on the client access server

- Approximately 130MB of message tracking logs on the hub transport server

- Approximately 60MB of transport protocol logs on the hub transport server

- Approximately 25MB of connectivity logs on the hub transport server

I generally like to keep 30 days worth of logs available for auditing and troubleshooting. Assuming that the logs are created equally 7 days a week (which they are not, of course), and I want to keep 30 days worth of logs, I want to ensure that I have a minimum of 7.2GB ((25*30) + (130*30) + (60*30) + (25*30)) of space available for all of these logs. This is, of course, assuming that the client access server and the hub transport server are on the same server role. This may seem like a trivial amount to worry about, but what if you have to scale a hub transport server or a client access server to support a user community of 25,000 mailboxes. Suddenly that is a significant amount of disk space.

Understanding IOPS

If you are supporting an Exchange Server with 50 or 75 mailboxes, you don't need to worry about the I/O load that the users put on the disk subsystem. However, once you start scaling to the hundreds of mailboxes (or higher), you need to start taking into considering not only the capacity of the disk subsystem but also whether it can keep up with the disk I/O load that the users are putting on it.

In the Exchange world, this load is normally known as the I/O per second load or just IOPS (pronounced eye-ops). Although the concept of sizing a disk system to support a minimum number of IOPS is certainly not an Exchange-only concept, your Exchange Servers (certainly the Mailbox servers) are usually the most consistently used servers in your environment.

If the disk subsystem is not sized correctly for I/O capacity, users will notice performance delays when sending and receiving mail. If it is significantly under-powered, not only will users see noticeable performance problems but also email may be delayed.

In the previous section of this chapter, we calculated the amount of storage required for 1000 fairly heavy users. We estimated that the worst-case amount of disk storage required is about 1.3TB. If we added another 500GB just for good measure, the disk space requirement is 1.8TB. The typical way some messaging professionals solve this is to put the largest disk drives they can find into a single RAID-5 array.

To get 1.8TB of available disk space, we could put seven 300GB drives in a single RAID-5 array. That would yield 1.8TB of useable disk space for those 1000 mailboxes. However, would that yield sufficient I/O capacity? Without going into a whole lot more detail just yet, under ideal circumstances, an Exchange 2003 server with 1000 heavy mail users would require approximately 1000 IOPS of maximum I/O capacity. Those same 1000 heavy mail users on an Exchange 2007 Mailbox server would require approximately 300 IOPS capacity. You might think that seven disks in a RAID-5 array would provide a lot of I/O capacity, but depending on the speed of the drives, this array may provide as little as 350 IOPS capacity. Still enough to handle 1000 Exchange 2007 mailboxes, but just barely. And we did not take into consideration other factors that might increase (or decrease) the IOPS load.

Before jumping into the world of I/O planning, I want to make clear that any type of I/O capacity planning requirements we make based either on Microsoft's recommendations or my own observations are based on averages. If we conclude that a specific user community has an I/O profile that requires 1.0 IOPS, this does not mean that ALL users are working at the same time. A single user sending a single message can generate 50 I/O requests in a single second. The estimates are based on the average usage of the user community.

Further, I am using a concurrency factor of 100 percent; meaning that all users are using the system during a typical period of operations; for most organizations, this is from approximately 7:00a.m. to 6:00p.m.. However, if you know that even though you support 1000 mailboxes but at any given time only 50 percent of those mailboxes are in use, you can reduce you I/O requirement estimates.

Microsoft Typical User Profile

The Exchange Team at Microsoft has done a lot of research on Exchange Servers, its use of the disk subsystem, and its I/O requirements. Over the years, the I/O profiles have continually changed; they have changed for a number of reasons including improvements in available memory for caching data, improvements in the ESE database engine, and the fact that more users are "power email users" today than there were even 3 or 4 years ago. A secretary or clerk that sent or received five emails a day 5 years ago may send and receive 40 emails today.

Over the years, Microsoft has published a series of blog entries and whitepapers to help you estimate just how much I/O a typical user community will generate. The company starts by first defining the type of user in terms of messages sent, messages received, and their mailbox size, then providing an estimated database IOPS load for the different user types. The values are different for Exchange 2003 and Exchange 2007 mailbox servers for reasons I will talk about in the next section.

Exchange 2003 IOPS Requirements

Let's first look at Exchange 2003 IOPS requirements. Table 2.1 shows the estimated IOPS requirements for four types of Exchange users. These IOPS estimates are just for estimating the database IOPS and does not include transaction log writes or other types of disk activity.

User Type	Mailbox Size	Messages Sent/Received Per Day	IOPS Requirements Per User
Light	50MB	20 sent / 50 received	.5
Average	100MB	30 sent / 75 received	.75
Heavy	200MB	40 sent / 100 received	1.0
Large	500MB	60 sent / 150 received	1.5

Table 2.1: Exchange 2003 user profiles and IOPS requirements.

For the organization with 1000 mailboxes, we are estimating that the typical user is a "heavy" user. Granted that there will be some "light" and "average" users in that organization, but there will also be a few users that are "large" email users (like me!); but we are guessing that the typical user is a "heavy" user. This means that from the disk subsystem that supports the Exchange Server 2003 mailbox databases, we will require approximately 1000 IOPS during peak periods.

Keep in mind that the figures are designed to guarantee that every user experiences sub-second response times from the mail server. Some organizations may not require instantaneous response for all their users all the time, so you can back off of these general guidelines as appropriate. For organizations that are using Outlook in local cache mode, sub-second response times are generally not necessary.

Microsoft recommends breaking this into the read and write I/O load so that you know how much of this load is "write" intensive. Disk writes are more disk intensive than reads. Knowing your read and write ratio can help you optimize the RAID controller or SAN caching configuration. Microsoft estimates that the read to write ratio is somewhere between 75/25 (four reads for each write) and 66/33 (three reads for each write) for an Exchange 2003 server. For estimation purposes, I recommend using the more write-intensive of the two, or 66/33. For Exchange 2007, the ratio is leaning even more towards write-intensive, although the overall I/O load is decreased.

Let's take a look at a real-world example of this (with the server name blanked out). Figure 2.10 shows the physical disk counters Disk Reads/sec, Disk Writers/sec, and Disk Transfers/sec. This Exchange 2003 server supports about 1000 mailboxes; this performance monitor data was logged from 1:30p.m. to 3:30p.m. one day during the middle of the week. The thing that I want to illustrate from this performance monitor screen is the read and write ratio on the database disks (E and F). There is an average of 215 disk transfers per second during this time. Out of that, 153 transfers per second are read operations or 71.9 percent and 62 transfers per second are write operations or about 28.8 percent.
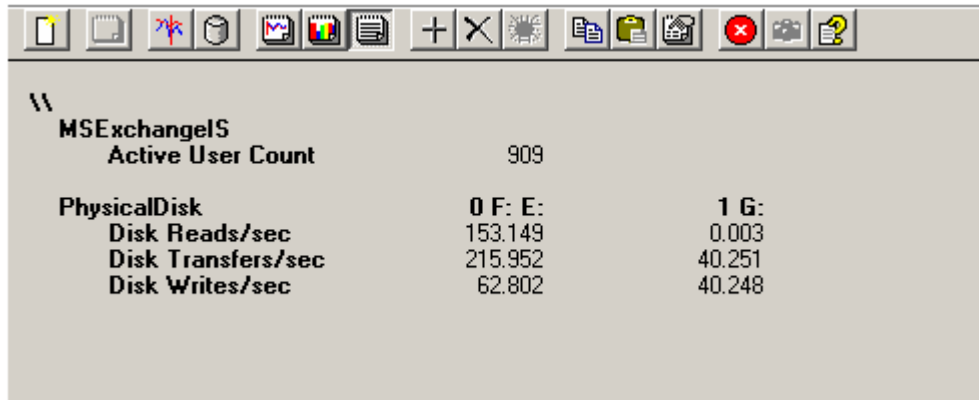
Figure 2.10: Examining the read and write request ratios.

The G drive holds the transaction logs for the databases that are on the E and F drives. Note the read versus write ratio for the transaction log disk. It is almost 100 percent write operations. Is that expected? You bet it is. In normal operations, the transaction log disk has almost no read operations. That will, of course, change with Exchange 2007 when you are using either local continuous replication (LCR) or CCR.

   📖 I will cover more on LCR and CCR later.

☞ Would you base your entire I/O capacity planning requirements on one 2-hour performance monitor log? No way! This particular performance log is not even from this organization's peak usage period. If I were planning for I/O capacity using this organization's logs, I would record key performance monitor counters during peak periods (usually about 8:00a.m. to 11:00a.m, each day for most organizations) and look at the values from each day. A single day's performance monitor logs are not sufficient for capacity planning.

Another interesting tidbit you may glean from this is the ratio of transaction log writes to the overall disk transfer/sec requirements. There was an average of 40 disk transfers per second required for the transaction log disk and 215 disk transfers per second.

In Figure 2.10, the transaction log I/O is approximately 19 percent of the I/O required for database operations. Microsoft recommends that for the transaction log writes that you plan for approximately 10% of the database I/O. Clearly in this case we are closer to 20 percent.

There was one other interesting piece of information that I wanted to point out from the performance monitor information in Figure 2.10. There was an average of 909 active users during this performance monitor session. In this particular user community, I rate most of their users as "average" user types; this means an I/O requirement of approximately .75 IOPS per user. Under typical circumstances, we would expect the I/O requirements to be 682 disk transfers per second (.75 IOPS per user * 909 active users), but the database drive's IOPS per second averaged only 215 transfers per second.

There are a few reasons for this, not the least of which is that the log file was recorded in the middle of the afternoon. The middle of the afternoon is not as busy for this organization as the morning. Another reason (and the point I am making) is that the Report view of performance monitor is an average over the entire length of the data collection period (in this case, 2 hours). The chart view will allow you to look at the activity generated over the period.

Figure 211 shows the chart view for the data previously presented in Figure 2.10, but I simplified it somewhat so that you only see the active connection counter and the disk transfers/sec counter. Further, I changed the scale on these two counters to 0.1 rather than 1 so that on the scale from 1 to 100, the counters will fit properly. When interpreting the data on the y axis (the vertical axis), you must multiply the y-axis value by 10 to get the actual value.
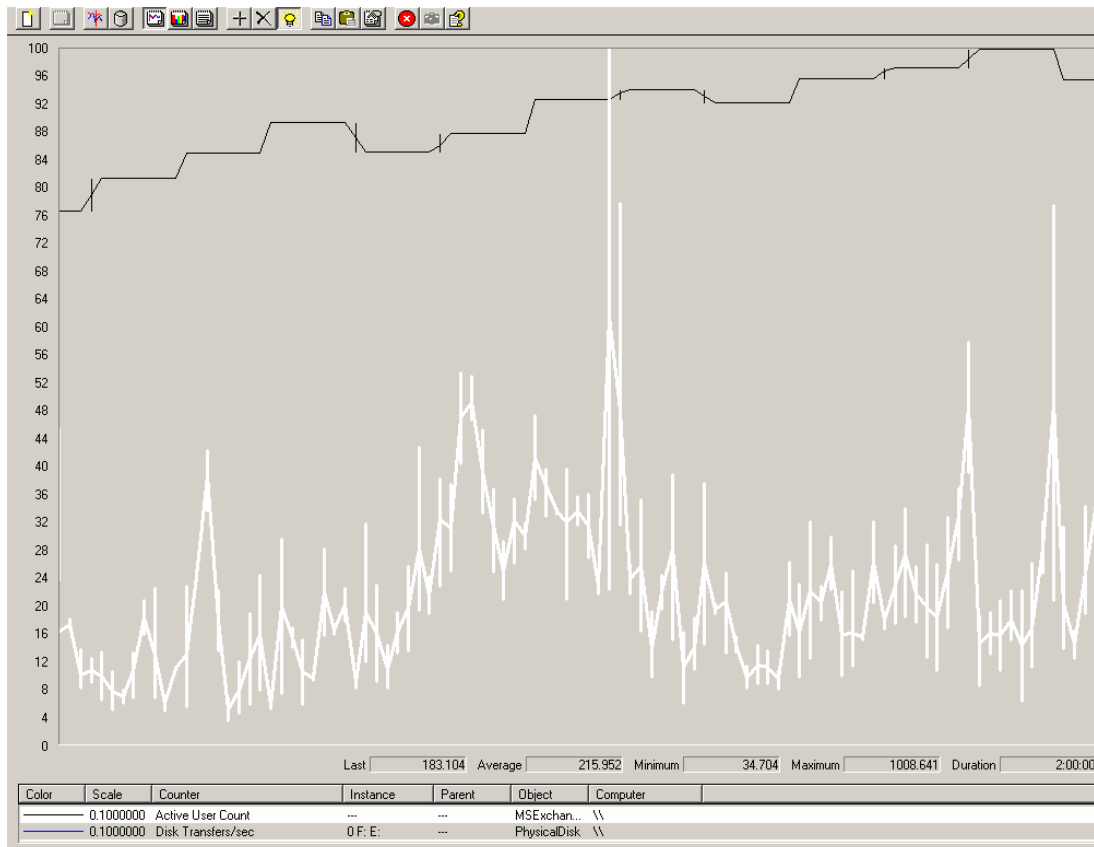
Figure 2.11: Viewing the disk transfers/sec over time.

Notice that the number of active users started at just over 760, but climbed upwards to over 1000 during the time the data was recorded. Also notice the disk I/O jumped all over the place. Since I have the Disk Transfer/sec counter highlighted on the bottom part of the screen, I can see the last, average, minimum, and maximum values. During the recorded log file, the minimum Disk Transfers/sec was only 34.7, but it did jump to 1008.6 right near the middle of the chart.

When you design for I/O capacity, look at a typical worst case. If you find frequent I/O jumps to nearly 1000 Disk Transfers/sec, you want to plan for at least 1000 Disk Transfers/sec capacity on the disk subsystem that supports the database disks.

The 1000 active mailbox user figure per mailbox server seems to be a sweet spot for Exchange 2003 and estimating IOPS capacity. That seems to be where the maximum available RAM for caching seems to be the most effective. Exchange 2003 allocates a maximum of 896MB of RAM for database caching by default or an absolute maximum amount of RAM for caching of 1.2GB of RAM with tweaking (see Microsoft Knowledge Base article 815372.) After about the 1000 user count, there are more simultaneous users competing for the same amount of memory cache and you reach a point of diminishing returns. Once you have more users, there is less "per user cache" and thus the I/O load on the disk subsystem increases pretty substantially. The actual additional overhead is about 25 percent. Table 2.2 shows the actual IOPS that are required when you increase the number of users above 1000 and assuming the estimated user profile is for "heavy" users.

Mailboxes	Estimated IOPS Per User	Calculated IOPS Required	IOPS Required With Diminished Cache Factor
1000	1.0	1000	1000
2000	1.0	2000	2500
3000	1.0	3000	3750
5000	1.0	4000	5000

Table 2.2: Estimating IOPS with diminished cache performance.

Exchange 2007 addresses this issue because there is more memory available for caching; this reduces the I/O load on the disk subsystem. However, if you built an Exchange 2007 server with less memory than is recommend, you will still pay the price in additional disk I/O.

Exchange 2007 IOPS Requirements

One of the biggest advances in Exchange Server 2007 is the reduction in the I/O requirements. Exchange 2007 achieves this by allowing more memory to be available for caching of data and thus is able to write to the disk more efficiently. More memory is available because Exchange 2007 is a 64-bit application rather than a 32-bit application. However, these additional gains in the I/O performance will not be realized if you do not follow the recommended memory guidelines. I will discuss the memory recommendations in the next section.

> ✎ Estimated IOPS capacity requirements are based on a server that has been running for at least a few hours. Every time the information store service is restarted, the caching engine has to start over and optimize itself. This cold-start state will require somewhat higher IOPS in situations such as after a server reboot, information store restart, cluster continuous replication failover, or single copy cluster failover. Do not assume that I/O measurements immediately after one of these events are valid for ongoing I/O requirements.

Microsoft has released some new statistics regarding IOPS requirement for Exchange Server 2007. The user profile (sent and received messages per day) is slightly different than the figures available from Microsoft for Exchange 2003, so it is difficult to compare Exchange 2003 and Exchange 2007 in an exact "apples for apples" comparison; however the reduction in I/O load (with sufficient memory for caching) is still up to a 70 percent reduction over Exchange 2003. These statistics also assume that the average message size is 50KB. Table 2.3 lists some of the Exchange Server 2007 IOPS requirements for different user profiles.

User Type	Database Cache Per User	Messages Sent/Received Per Day	IOPS Requirements Per User
Light	2MB	5 sent/20 received	0.11
Average	3.5MB	10 sent/40 received	0.18
Heavy	5MB	20 sent/80 received	0.32
Very heavy	5MB	30 sent/120 received	0.48

Table 2.3: Exchange 2007 IOPS requirements for different types of users.

Transaction Log IOPS Requirements

The database transaction log I/O requirements are not nearly as intensive as the IOPS requirements for the actual databases. Microsoft estimates that the transaction log IOPS requirements should be about 10 percent of the database IOPS requirements. In the example of an Exchange 2003 server's Disk Transfers/sec for the databases versus the transaction logs, we saw in performance monitor that the IOPS for the transaction logs was actually closer to 20 percent.

I recommend as a planning figure for transaction log IOPS requirements using 25 percent of the total database IOPS requirements. This will ensure you have sufficient capacity for the transaction log I/O requirements.

If you are using continuous replication technology (LCR or CCR), you must add 10 percent IOPS requirements above and beyond the base transaction log IOPS recruitments. For example, you determine that you require 200 IOPS for transaction log I/O; you need to factor another 20 IOPS on top of that for the additional read IOPS requirements.

Memory and IOPS Requirements

Sizing memory for Exchange 2003 servers is pretty simple. For a server with more than 100 mailboxes, simply add 4GB of RAM and you are done. Exchange 2003 does not scale past about 4GB of RAM. However, sizing the correct amount of RAM for an Exchange 2007 server is a bit more complex. Microsoft recommends all server roles have a minimum of 2GB of physical memory, but in my experience thus far I am recommending a minimum of 4GB.

Table 2.4 shows the memory recommendations based on the server roles. Notice that there is a maximum recommended RAM per server. This is because for each server, you reach a point of diminishing returns when it comes to adding memory; Exchange simply cannot effectively use the RAM past the maximum recommended amount. The amounts in Table 2.4 also assume that the server is running only the specific server role listed unless otherwise stated. I have also included the recommended and maximum recommend CPU cores in Table 2.4 for reference. I left out the minimum number of CPU cores because that the minimum is one CPU core for each type of server role.

Server Role	Minimum Recommend RAM	Recommended RAM Per CPU Core	Maximum Recommended RAM	Recommended CPU Cores	Maximum CPU Cores
Client Access	2GB	1GB per CPU core	8GB/Server	4 CPU cores	4 CPU cores
Combined function (hub transport, client access, unified messaging, and/or mailbox)	2GB	4GB + 2MB to 5MB per mailbox	8GB/Server	4 CPU cores	4 CPU cores
Edge transport	2GB	1GB per CPU core	16GB/Server	4 CPU cores	4 CPU cores
Hub transport	2GB	1GB per CPU core	16GB/Server	4 CPU cores	8 CPU cores
Mailbox	2GB	4GB + 2MB to 5MB per mailbox	32GB/Server	4 CPU cores	8 CPU cores
Unified messaging	2GB	1GB per CPU core	4GB/Server	4 CPU cores	4 CPU cores

Table 2.4: Recommended memory and CPU configuration.

One interesting thing you may note in this table is that the combined function or multi-role server (a server that supports more than one role) actually has less recommended maximum RAM and CPU cores than a mailbox server or hub transport server. The logic behind this is that at a certain point (such as past 500 or 1000 mailboxes), you should split server roles in to multiple physical servers rather than combining all functions onto a single system.

> 💣 If you do not provide the minimum recommend RAM for the user type that fits your user community, IOPS requirements will be higher than stated here. If you have not planned for this in your disk I/O capacity, performance will suffer.

Notice also that the server roles that support mailboxes include a calculation for additional memory for each mailbox supported. This additional memory calculation depends on the types of users you have in your environment (light, average, heavy). Table 2.5 shows the additional memory recommendations based on specific user types. By adding memory for each mailbox user, you increase the available cache, improve disk I/O performance, and reduce the overall IOPS requirements from the disk subsystem.

User Type	Additional Memory Per User
Light	2MB per mailbox
Average	3.5MB per mailbox
Heavy	5MB per mailbox

Table 2.5: Additional RAM for Exchange 2007 based on user type.

The maximum recommend RAM for any Exchange Server 2007 server is 32GB. This is because you reach a point of diminishing returns. It is not that Windows 2003 x64 and Exchange 2007 will not use the memory, but that the cost of more than 32GB of RAM versus the cost of adding disks to support additional IOPS requirements will be much higher. It is a lot cheaper to simply buy more disks to support the additional IOPS requirements than it is to buy more than 32GB of RAM. This will more than likely change with newer generations of servers.

☞ If you have a mailbox server that has local continuous replication, add 1GB of additional memory to your memory requirements.

There is an alternative method to calculating the minimum memory that mailbox servers should require. This is to factor in the number of storage groups that the mailbox server supports. Table 2.6 shows the minimum estimated RAM for an Exchange Server 2007 mailbox server given a specific number of storage groups. When you calculate the amount of RAM required based on per mailbox counts or per storage group counts, you should take the larger of the two values as the amount of RAM you should use.

Number of storage groups	Minimum RAM required
1 - 4	2GB
5 - 8	4GB
9 - 12	6GB
13 - 16	8GB
17 - 20	10GB
21 - 24	12GB
25 - 28	14GB
29 - 32	16GB
33 - 36	18GB
37 - 40	20GB
41 - 44	22GB
45 - 48	24GB
49 - 50	26GB

Table 2.6: RAM required based on the number of storage groups.

So now let's summarize the amount of RAM that we would require for the 1000 "heavy" user Exchange organization. We estimated that we would require 13 mailbox databases in 13 separate storage groups. This means the minimum RAM required is 8GB. Table 2.5 indicates we should add 5MB of RAM for each mailbox; this comes up to 5GB of RAM for the mailboxes plus a minimum of 2GB of RAM for the server for a total of 7GB of RAM recommended. Therefore, we should use 8GB of RAM because that is the minimum amount recommended for 13 storage groups.

Factors Affecting I/O Requirements

Besides just user load, there are other factors that may affect IOPS requirements. Of course, most of these factors impact IOPS requirements in a negative way. In the above calculations, we have not taken additional factors into consideration. The following list highlights some of the factors and how the factor may affect the IOPS load:

- Third-party remote mail clients such as GoodMail and BlackBerry increase IOPS requirements for each simultaneous client. By some estimates, one BlackBerry device can place a load on the server equal to four MAPI clients.

- Antivirus and anti-spam software that reside on mailbox servers can increase the I/O load if they are performing real-time scanning operations. This also true of anti-spam and antivirus application that reside on hub transport servers and perform transport scanning of email. Scheduled antivirus scans on mailbox servers can increase the I/O load and should be performed during off-hours only if they are necessary.

- Configuring Windows Mobile and ActiveSync devices to synchronize frequently to the Exchange Server (instead of using "push email") can increase the I/O load for each mailbox to three or four times that of what a typical MAPI client generates.

- Streaming backups increase the I/O load on mailbox servers. Streaming backups should be performed during off-peak hours if possible. If streaming backups must be performed during normal periods of operation, consider performing differential or incremental backups during peak periods and full backups during off-peak hours.

- Using Outlook 2003 or Outlook 2007 in local cache mode can decrease the IOPS requirements because users are always working on the local copy of their mailbox.

- Exchange 2007 full text indexing can increase the IOPS requirements for the database by approximately five percent. Keep in mind that Exchange 2007 has full text indexing enabled for all databases by default.

- Messaging records management and online maintenance operations increase the IOPS requirements for the mailbox databases. These operations should be scheduled during off-peak hours.

- CCR and LCR can increase transaction log IOPS requirements by an additional 10 percent on the active transaction log volume.

- LCR replicates data to another physical disk or LUN on the same physical Exchange 2007 mailbox server. The passive volumes (the location where the transaction logs are copied and then replayed) should not be on the same volumes as the active transaction logs and databases. If these volumes are LUNs on a SAN, be sure to include the IOPS requirements when planning for IOPS capacity on your SAN. Based on one estimate I have read, the IOPS requirements for the passive LUNs can be as much as twice what the active LUNs require.

- RAID5, LUN replication, and snapshots all consume IOPS. These "in-SAN" IOPS are tricky because they don't show up in the traditional server monitoring tools. For these, you must rely on the SAN management tools and monitoring software.

Sizing SANs for Sufficient I/O Capacity

For most of us, what goes on inside a SAN is very much like a black-box. Data goes in and we are able to retrieve data out. The "old school" way of thinking about sizing and partition disks for Exchange Server (such as putting transaction logs on separate spindles from the databases) does not usually work with SANs. You may actually get a very weird look from your SAN administrator if you ask for a LUN for transaction logs that is on a separate RAID-1 array from the database files.

SANs are designed to be both flexible and scalable, so it is not a common occurrence for a SAN administrator to carve out separate physical disks for different LUNs. If this is even possible with your particular SAN vendor, it might make the overall management of the SAN less scalable or flexible and doing so could hurt the performance capacity that was originally designed for the SAN.

One key attribute you should look for with a SAN is "ease of scaling." Given that performance sizing is complex, you want a storage system where it is very easy to scale performance (that is, increase IOPS) without re-designing or re-layouts of LUNs. A key question to ask is, "If we need another 200 IOPS for this LUN, how quickly/easily could we do this?" Ideally, you should be able to do this on the fly in a matter of minutes.

Instead of asking your SAN administrator to carve the SAN into a lot of separate RAID-1, RAID-5, or RAID 0+1 LUNs, instead plan for the total I/O capacity required for all the transaction logs, databases, message tracking logs, protocol logs, index files, LCR transaction logs, LCR databases, and so on and then add an additional factor of 20 to 30% to provide you some headroom. Provide your total IOPS requirements to the person that will size and configure your SAN. Be prepared to answer questions about factors such as minimum and maximum LUN size as well as how much room you need to grow.

Determining Theoretical IOPS Maximum Capacity

Microsoft provides some tools for sizing and testing both disk and server capacity. These can help you measure existing capacity or confirm that your planned capacity and disk subsystem will actually support the expected loads. These Exchange capacity tools can be downloaded from http://technet.microsoft.com/en-us/exchange/2007/bb330849.aspx and include the following:

- The JetStress utility is a tool that you configure to emulate the operation of an Exchange database. JetStress writes transaction logs and reads/writes simulated database files and allows you to measure I/O capacity on your disk subsystem based on the parameters you use when you set up JetStress.

- The Exchange Server Profile Analyzer collects statistical information from a single mailbox database or many mailbox database. This data can be used to analyze the help of an Exchange Server.

- The Exchange Load Generator performs stress testing tasks, pre-deployment validation, and benchmarking tests by introducing different types of loads and simulating MAPI client access in a test Exchange system. This tool was formerly known as LoadSim or Load Simulator.

- The Exchange Server Stress and Performance tool is similar to the Load Generator except that it helps to simulate multiple Internet protocols (HTTP, POP3, IMAP4) against an Exchange Server 2003 front-end server or an Exchange Server 2007 client access server.

There are both 32-bit and 64-bit versions of these tools, so make sure you get the right version for your particular platform. Do not run these tools on production systems.

Summary

Sizing disk storage to support Exchange Servers is more than just tossing a bunch of disks at the server and hoping they are sufficient. You must not only calculate the number of disks that are required to provide adequate storage but also ensure that they will collectively provide sufficient I/O capacity.

Exchange Server databases must be sized on additional factors other than just how big the mailbox will grow; the total size of Exchange data includes not only the mailbox data but also database white space and deleted items. If full text indexing is enabled, the full text indexing files should also be considered in disk space calculations.

The I/O load that the user community places on the disk subsystem must also be considered when sizing the disk subsystem. The I/O load may actually require more disks than the space consumed by the users' data. This was certainly the case in Exchange 2003 where a small number of disks may be required for the data but a larger number of disks are required to support the total IOPS load that the user community actually generates.

Taking a good, realistic look at your mail storage needs now and for the foreseeable future will help you to properly size your Exchange Servers. Sizing Exchange Server systems properly for not only available disk space but also disk I/O capacity helps to ensure that the Exchange Server operates trouble free and adequately servers your user community.

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit http://nexus.realtimepublishers.com.

Realtime
publishers
"Leading the Conversation"