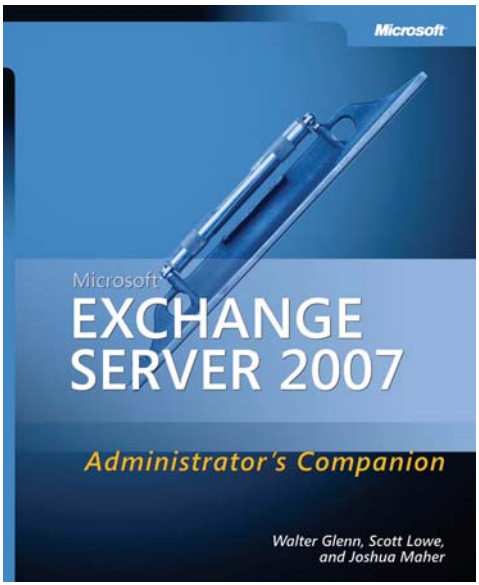



# Microsoft® Exchange Server 2007 Administrator's Companion

*Walter Glenn, Scott Lowe, and Joshua Maher*

To learn more about this book, visit Microsoft Learning at
http://www.microsoft.com/MSPress/books/9545.aspx

9780735623507

Microsoft Press

# Resources for IT Professionals

*Published and Forthcoming Titles from Microsoft Press*

## Windows Server

**Microsoft® Windows Server® 2003 *Resource Kit***
Microsoft MVPs and Partners with Microsoft Windows Server Team
978-0-7356-2232-6

**Microsoft Windows Server 2003 *Administrator's Companion* Second Edition**
Charlie Russel, Sharon Crawford, and Jason Gerend
978-0-7356-2047-6

**Microsoft Windows Server 2003 *Inside Out***
William R. Stanek
978-0-7356-2048-3

**Microsoft Windows Server 2003 *Administrator's Pocket Consultant* Second Edition**
William R. Stanek
978-0-7356-2245-6

## Windows Client

**Windows Vista™ *Resource Kit***
Tulloch, Northrup, Honeycutt, Russel, and Wilson with the Microsoft Windows Vista Team
978-0-7356-2283-8

**Windows Vista *Administrator's Pocket Consultant***
William R. Stanek
978-0-7356-2296-8

**Microsoft Windows® XP Professional *Resource Kit* Third Edition**
The Microsoft Windows Team with Charlie Russel and Sharon Crawford
978-0-7356-2167-1

**Microsoft Windows XP Professional *Administrator's Pocket Consultant* Second Edition**
William R. Stanek
978-0-7356-2140-4

**Microsoft Windows Command-Line *Administrator's Pocket Consultant***
William R. Stanek
978-0-7356-2038-4

## SQL Server 2005

**Microsoft SQL Server™ 2005 *Administrator's Pocket Consultant***
William R. Stanek
978-0-7356-2107-7

**Microsoft SQL Server 2005 *Administrator's Companion***
Whalen, Garcia, et al.
978-0-7356-2198-5

**Inside Microsoft SQL Server 2005: The Storage Engine**
Kalen Delaney
978-0-7356-2105-3

**Inside Microsoft SQL Server 2005: T-SQL Programming**
Itzik Ben-Gan, Dejan Sarka, and Roger Wolter
978-0-7356-2197-8

## Exchange Server 2007

**Microsoft Exchange Server 2007 *Administrator's Companion***
Walter Glenn and Scott Lowe
978-0-7356-2350-7

**Microsoft Exchange Server 2007 *Administrator's Pocket Consultant***
William R. Stanek
978-0-7356-2348-4

## Scripting

**Microsoft Windows PowerShell™ *Step by Step***
Ed Wilson
978-0-7356-2395-8

**Microsoft VBScript *Step by Step***
Ed Wilson
978-0-7356-2297-5

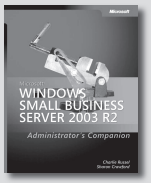
**Microsoft Windows Scripting with WMI: Self-Paced Learning Guide**
Ed Wilson
978-0-7356-2231-9

**Advanced VBScript for Microsoft Windows Administrators**
Don Jones and Jeffery Hicks
978-0-7356-2244-9

## RELATED TITLES


**Microsoft Office SharePoint® Server 2007 *Administrator's Companion***
Bill English with the Microsoft SharePoint Community Experts
978-0-7356-2282-1


**Microsoft Windows Security *Resource Kit* Second Edition**
Ben Smith and Brian Komar with the Microsoft Security Team
978-0-7356-2174-9




**Microsoft Windows Small Business Server 2003 R2 *Administrator's Companion***
Charlie Russel and Sharon Crawford
978-0-7356-2280-7

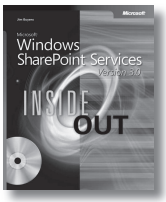

**Microsoft Internet Security and Acceleration (ISA) Server 2004 *Administrator's Pocket Consultant***
Bud Ratliff and Jason Ballard with the Microsoft ISA Server Team
978-0-7356-2188-6

# 2007 Microsoft® Office System Resources for Developers and Administrators


### Microsoft Office SharePoint® Server 2007 Administrator's Companion

Bill English with the Microsoft SharePoint Community Experts
ISBN 9780735622821

Get your mission-critical collaboration and information management systems up and running. This comprehensive, single-volume reference details features and capabilities of SharePoint Server 2007. It delivers easy-to-follow procedures, practical workarounds, and key troubleshooting tactics—for on-the-job results.




### Microsoft Windows SharePoint Services Version 3.0 Inside Out

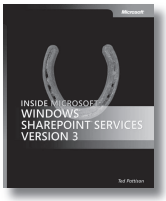
Errin O'Connor
ISBN 9780735623231

Conquer Microsoft Windows SharePoint Services—from the inside out! This ultimate, in-depth reference packs hundreds of time-saving solutions, troubleshooting tips, and workarounds. You're beyond the basics, so now learn how the experts tackle information sharing and team collaboration—and challenge yourself to new levels of mastery!


### Microsoft SharePoint Products and Technologies Administrator's Pocket Consultant

Ben Curry
ISBN 9780735623828

Portable and precise, this pocket-sized guide delivers immediate answers for the day-to-day administration of Sharepoint Products and Technologies. Featuring easy-to-scan tables, step-by-step instructions, and handy lists, this book offers the straightforward information you need to get the job done—whether you're at your desk or in the field!




### Inside Microsoft Windows® SharePoint Services Version 3

Ted Pattison and Daniel Larson
ISBN 9780735623200

Get in-depth insights on Microsoft Windows SharePoint Services with this hands-on guide. You get a bottom-up view of the platform architecture, code samples, and task-oriented guidance for developing custom applications with Microsoft Visual Studio® 2005 and Collaborative Application Markup Language (CAML).

### Inside Microsoft Office SharePoint Server 2007

Patrick Tisseghem
ISBN 9780735623682

Dig deep—and master the intricacies of Office SharePoint Server 2007. A bottom-up view of the platform architecture shows you how to manage and customize key components and how to integrate with Office programs—helping you create custom enterprise content management solutions.

### Microsoft Office Communications Server 2007 Resource Kit

Microsoft Office Communications Server Team
ISBN 9780735624061

Your definitive reference to Office Communications Server 2007—direct from the experts who know the technology best. This comprehensive guide offers in-depth technical information and best practices for planning, designing, deploying, managing, and optimizing your systems. Includes a toolkit of valuable resources on CD.

### Programming Applications for Microsoft Office Outlook® 2007

Randy Byrne and Ryan Gregg
ISBN 9780735622494

### Microsoft Office Visio® 2007 Programming Step by Step

David A. Edson
ISBN 9780735623798

*See more resources at* **microsoft.com/mspress** *and* **microsoft.com/learning**

Microsoft Press® products are available worldwide wherever quality computer books are sold. For more information, contact your bookseller, computer retailer, software reseller, or local Microsoft Sales Office, or visit our Web site at **microsoft.com/mspress**. To locate a source near you, or to order directly, call 1-800-MSPRESS in the United States. (In Canada, call **1-800-268-2222**.)

Microsoft® Press

# Windows Vista™ Resources for Administrators


### Windows Vista Administrator's Pocket Consultant

William Stanek
ISBN 9780735622968

Portable and precise, this pocket-sized guide delivers immediate answers for the day-to-day administration of Windows Vista. Featuring easy-to-scan tables, step-by-step instructions, and handy lists, this book offers the straightforward information you need to solve problems and get the job done—whether you're at your desk or in the field!


### Windows Vista Resource Kit

Mitch Tulloch, Tony Northrup, Jerry Honeycutt, Ed Wilson, Ralph Ramos, and the Windows Vista Team
ISBN 9780735622838

Get the definitive reference for deploying, configuring, and supporting Windows Vista—from the experts who know the technology best. This guide offers in-depth, comprehensive technical guidance on automating deployment; implementing security enhancements; administering group policy, files folders, and programs; and troubleshooting. Includes an essential toolkit of resources on DVD.


### MCTS Self-Paced Training Kit (Exam 70-620): Configuring Windows Vista Client

Ian McLean and Orin Thomas
ISBN 9780735623903

Get in-depth preparation plus practice for Exam 70-620, the required exam for the new Microsoft Certified Technology Specialist (MCTS): Windows Vista Client certification. This 2-in-1 kit focuses on installing client software and configuring system settings, security features, network connectivity, media applications, and mobile devices. Ace your exam prep—and build real-world job skills—with lessons, practice tests, evaluation software, and more.

### MCITP Self-Paced Training Kit (Exam 70-622): Installing, Maintaining, Supporting, and Troubleshooting Applications on the Windows Vista Client – Enterprise

Tony Northrup and J.C. Mackin
ISBN 9780735624085

Maximize your performance on Exam 70-622, the required exam for the new Microsoft® Certified IT Professional (MCITP): Enterprise Support Technician certification. Comprehensive and in-depth, this 2-in-1 kit covers managing security, configuring networking, and optimizing performance for Windows Vista clients in an enterprise environment. Ace your exam prep—and build real-world job skills—with lessons, practice tests, evaluation software, and more.

### MCITP Self-Paced Training Kit (Exam 70-623): Installing, Maintaining, Supporting, and Troubleshooting Applications on the Windows Vista Client – Consumer

Anil Desai with Chris McCain of GrandMasters
ISBN 9780735624238

Get the 2-in-1 training kit for Exam 70-623, the required exam for the new Microsoft Certified IT Professional (MCITP): Consumer Support Technician certification. This comprehensive kit focuses on supporting Windows Vista clients for consumer PCs and devices, including configuring security settings, networking, troubleshooting, and removing malware. Ace your exam prep—and build real-world job skills—with lessons, practice tests, evaluation software, and more.

---

*See more resources at* **microsoft.com/mspress** *and* **microsoft.com/learning**



Microsoft® Press

# Table of Contents

Part IV
# Management

Chapter 16

# Disaster Recovery



Backing up and restoring Microsoft Exchange Server 2007 databases are critically important aspects of Exchange planning and configuration. Unfortunately, many organizations overlook the importance of backing up and restoring Exchange servers, and even when they do perform regular backups, they may not test those backups appropriately.

This chapter focuses on the backup and recovery of your Exchange Server 2007 databases. The first part of the chapter details the Exchange database architecture, the types of backups, and the types of restores. The second part of the chapter discusses methods for implementing common backup and restore strategies. You also become familiar with the tools required to help implement and troubleshoot backup and restore issues in most situations.

## Backup and Restore Technologies

This section introduces the Exchange database architecture and the types of backups and restores that are possible within that architecture. Several Exchange Server 2007 features are mentioned, such as Local Continuous Replication (LCR) and Clustered Continuous Replication (CCR). These are log shipping features implemented by seeding a replica copy of the database on separate storage (LCR uses separate storage on the same server, while CCR uses separate storage on a additional cluster node) and replaying the closed transaction logs from the production copy into the seeded replica to keep it up to date.

## The Exchange Database

The core component of the Exchange Mailbox server role is the Exchange Information Store. Understanding the Information Store and the underlying Microsoft Extensible Storage Engine (ESE) database is an important first step to understanding how backups and restores work in Exchange Server 2007.

> **Note** The ESE database has previously been referred to as Jet Blue (which is a different version than the Jet Red database used by Microsoft Office Access).

### Basic Architecture

The ESE database in use by Exchange Server 2007 is the same version of the B+-Tree database used by Exchange Server 2003 SP1 and Active Directory. Exchange Server 2007 implements this database with an updated set of attributes including:

- The log file size changes from 5 MB to 1 MB.
- The database page size changes from 4 KB to 8 KB.

These attributes are necessary to support the built-in log shipping capabilities and support a lower I/O profile. The log shipping capabilities (Local Continuous Replication and Clustered Continuous Replication) require the lower log file size to break up the data loss potential into smaller chunks. The lower I/O profile is achieved by a number of features in Exchange Server 2007 and allows more users per server then previous versions. The attributes are important to configuration and performance, and in gaining an understanding of what happens during a backup or restore.

### Transactions

The database transactions are an ACID operation. ACID database transactions ensure integrity by being Atomic, Consistent, Isolated, and Durable.

- **Atomic** Indicates that a transaction state change is all or none, which means the entire transaction must be completed before any one part of the transaction can be recognized as completed. Atomic state changes include database page rearrangements, mailbox folder view additions, and e-mail message submissions. Without an Atomic nature, it is impossible to assure complete transactions.
- **Consistent** Indicates that a transaction is a correct transformation of the current state of the database. The actions taken as a group do not violate any one of the integrity constraints associated with the current state of the database. Without Consistent properties, it would be possible for corrupt e-mail messages to enter the database during regular operation.

- **Isolated** This term indicates that even though transactions run at the same time, it appears to each transaction (T) that others executed either before T or after T, but not both. Without Isolated properties, an item could be marked as read before the item is delivered to the mailbox.
- **Durable** This term indicates that as soon as a transaction is completed successfully (the commit operation enters the database), its changes survive failures. This also means that if a transaction does not complete in its entirety (no commit operation is specified for the transaction), the entire transaction is rolled back. Without Durable properties, the database would not be recoverable to the last e-mail message delivered to a mailbox through power failures, server outages, or other inconsistent states.

These properties are critical to backup and restore operations. Without them, an Exchange administrator would not enjoy the feeling of safety that comes with the infrequent corruption of ESE databases. These properties help guarantee the following rules:

- Exchange rolls back any changes (or e-mail messages) that are not received by the database in their entirety.
- Exchange disregards all pages that are not in order to prevent corruption.
- Exchange does not accept any operations that do not allow the database to easily become consistent.
- Exchange allows only one transaction at a time to be entered into the database, even though multiple transactions are accepted at once to allow for increased performance.
- Exchange guarantees that once a transaction is committed, the transaction is fully recoverable within the database file itself.

These guarantees are important to keep in mind when evaluating any backup/restore and disaster recovery strategy.

## Logging

So where do all of the log files come in? The basic principle behind the ESE database is that writing to memory is a cheaper operation than writing to disk. It has been this way since the inception of Exchange Server and is extended even further in Exchange Server 2007 with the move to a 64-bit architecture. The problem with writing to memory and later flushing those writes to disk is the information stored in memory is stateless; for Exchange, this means the e-mail message state is not guaranteed to be committed and recoverable while it is in memory. To ensure that the statelessness is not a problem, the

log files were built to ensure that all transactions are in the log file at the same time they are posted to memory. This is called a *dual-phase commit* and is illustrated in Figure 16-1.

- Phase 0: commit the user transaction in fast way

  Sequential write of page changes (modification, deletion, insertion)

- Phase 1: update the database in an atomic way

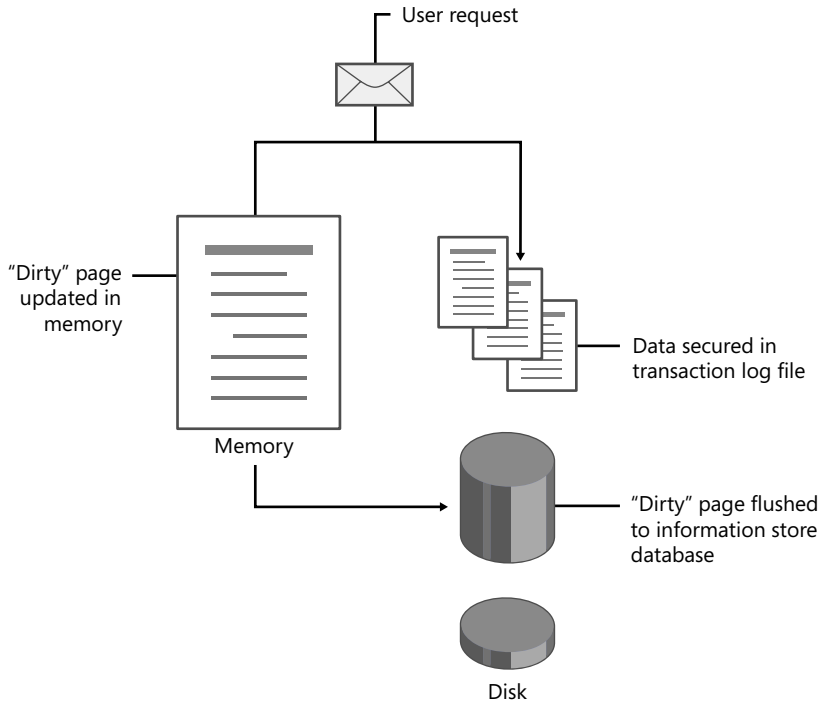



**Figure 16-1** Dual-phase commit

To illustrate the process, assume that User1 sends a 2500-KB (2.4-MB) message to User2, another user on the same message store:

- User1 sends a 2500-KB (2.4-MB) message.
- 312.5 8-KB pages are consumed in memory.
- 2.44 log files are written to disk.
- User1's message is registered in User1's Outlook client as Sent.
- User2 receives a message with a pointer to the record in memory for the 312.5 pages that contain the message.

At this point, the message is sent; it is in memory and has been written sequentially to a log file. With the message in the log files, it is in a semirecoverable state. It is only semirecoverable because the last part of the message (the .44 MB) is in an open log file. Additionally, the message is not stored or indexed in a way that the message is easily retrievable at a later date. For that to happen, the message must be written to the database. The ESE database has several ways to write data from memory to the database. Those methods are as follows:

- **Anomalous writes** This is the most common write that ESE performs. This is the scenario in which there is a page that has been entered into memory and not been requested lately. Such a page is often referred to as a dirty page.
- **Idle writes** This is the least common write that ESE performs. This is the scenario in which no other activity is happening on the server and there are many extra cycles to write data from memory to disk.
- **Opportune writes** In Exchange Server 2007, this is more common than in previous versions. The opportune writes are writes of pages that may not be ready to be written but are destined for database pages that are next to a write that is ready to be written. This could consist of several separate e-mails destined for a single B+-Tree, or two attachments destined for an attachment table, and so on.
- **Normal writes** Oddly enough, these are not normal. The normal write occurs when the checkpoint depth has reached its limit (which defaults to 20 MB per storage group). This scenario occurs only in a heavily loaded system and should be watched for. It also means that backups slow down during this time as the database must be in a recoverable state prior to taking a backup copy.
- **Repeatedly written** This does not commonly occur. It is only in heavily loaded systems that a page is repeatedly written. Repeatedly written means that a page has been entered into memory. That page is then written to disk via one of the four previous methods. After the page is written, the checkpoint location moves past that page. At this point, if the page is rediscovered, which would require a user to have immediately changed (edited, deleted, and so on) a message, it is recognized as a repeatedly written page.

It is important to understand these concepts to know how the ACID properties are implemented in the database and how different technologies may conflict with the nature of the ESE database. For example, a technology that backs up data in memory is not a good technology as those pages may be updated, removed, or rejected before they ever make it into the database.

## Circular Logging

*Circular logging* is intended to reduce storage requirements for the transaction logs after the transactions in the logs are committed to the databases. Circular logging is generally not recommended on production mail systems. Circular logging is used by default in ESE implementations where recoverability of a single database is not absolutely critical (for example, Active Directory or Exchange Hub Transport). This is due to the way circular logging treats the log files after they are committed. When circular logging is enabled, logs are removed from the system after they are committed to the database. This means there are only a few logs left on the system at any given time. This also means that during a restore, there is no way to roll forward the database as all logs are not backed up during a full backup. Fortunately, circular logging is disabled by default except on Edge and Hub Transport server roles. These server roles contain mostly transient data and under most circumstances do not require the databases to have backups performed on them.

## Checksum

The *checksum* (also called a message hash) is a string that is calculated and then added to each page in the database to verify page's integrity. The checksum itself does not guarantee data integrity; instead, the recalculation of the checksum when the page is read into RAM ensures that the data being read from the database is identical to the data that was written to the database.

When a page is loaded into RAM, the checksum is calculated and the page number is verified. If the checksum doesn't match the one that was written to the page when the page was written to the database, you can be sure that the page is damaged or corrupted. ESE ignores and/or corrects simple *bit flip* errors; these are errors where a single bit is written as a 1 instead of a 0. ESE ignores the error when it is found in conjunction with manual checksum verification (as in a checksum initiated during a Volume Shadow Copy Service [VSS] backup).

Note that ESE does not cause the damage to the page—it merely reports the damage to you. In nearly all instances, corruption to the database is the result of a hardware device or a device driver malfunctioning. ESE cannot cause page-level corruptions. These corruptions occur when the data is written to the disk and are caused by your hardware or device drivers. This is why it is imperative that you ensure all your firmware and device drivers are using the latest patches and updates and all hardware you are using is on the WHQL. Microsoft Customer Service and Support (CSS) will work with your hardware manufacturer to resolve any problems that might exist between your hardware and your Exchange Server 2007 database.

## Volume Shadow Copy Service

The Volume Shadow Copy Service (VSS) is a common method in use to back up and restore Exchange Server 2007. All VSS-based backups are considered online backups as they require the Exchange Information Store to be started during the process. VSS backups rely heavily on the Virtual Disk Service (VDS) and the Windows VSS framework. The premise behind the VSS framework is

- Windows provides a framework that governs how applications are backed up. Windows can do this because it understands all components that work under the Windows operating system.
- Applications provide writers that regulate when the application is ready to have a backup copy taken. The Application writer can do this because it understands the application.
- Microsoft and third-party vendors provide requestors that interact with an overall backup application. The requestor can do this because it understands how the backup application functions and it knows what data the backup application requires to be successful.
- Microsoft and third-party vendors provide hardware and software that understand how storage arrays can synchronize and subsequentially split the volumes.

The end result of these four components is a solution where a requestor asks Windows to set up an environment that supports a VSS snapshot to be taken. Once this is done, the requestor asks the application writer to set up a snapshot of the application. The writer then asks the provider to set up a snapshot based on whatever technology the provider supports. When all of the puzzle pieces are in place, the writer lets the requestor know when it has decided the application is ready for a snapshot to be taken. For Exchange, this means that the pages that are in the process of being written to disk have completed, and no new transactions are started in the database. This pause in activity is allowed to last for only 10 seconds while the provider takes its snapshot of the data. When the snapshot is complete, the requester informs the writer that everything went well and that the transaction processing can continue. At this point, the provider has obtained a copy of the database and log files. The provider then works with the requestor to ensure the database is checked for consistency and the logs on the production volume are truncated. To help visualize this, Figure 16-2 is a diagram of the process.

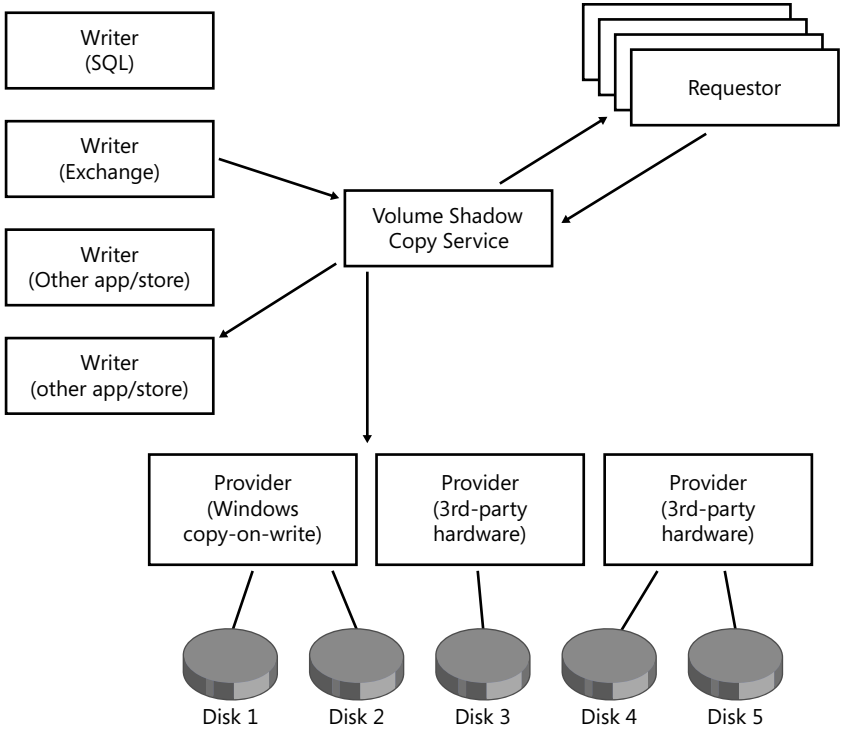



**Figure 16-2** Volume Shadow Copy Service (VSS) framework

## Types of VSS Backups Supported

The requestors available vary in functionality and in what VSS features are supported. For example, the VSS framework supports all of the standard backup methods: full, differential, incremental, and copy; however, most vendors implement only full and differential backups. This section describes what happens during each of these events.

### Full Backup

The database volume or file is mirrored to an alternate location via a hardware or software mechanism.

1. The Exchange writer informs the requestor when it is ready for a snapshot to be taken.
2. The requestor has 10 seconds to complete the snapshot. This includes mirroring the remaining blocks in the database or database volume that are not in sync and severing the relationship.
3. The requestor is then responsible for mounting the mirror to an alternate location to perform a checksum of the database. The checksum is run through a checksum

API or through the `eseutil.exe /k` command. Depending on the size of the database, the checksum process can be lengthy and demanding on the storage subsystem with a series of sequential reads. This process can be throttled to relieve the demand on the system; however, it still must complete prior to truncating the log files.

### Incremental Backup

The log volume or file is mirrored to an alternate location via a hardware or software mechanism.

1. The Exchange writer informs the requestor when it is ready for a snapshot to be taken.
2. The requestor has 10 seconds to complete the snapshot. This includes mirroring the remaining log file or blocks in the log file disk that are not in sync and severing the relationship.
3. The database files are left alone to be backed up during the next full backup. This means that each incremental backup that is taken contains all of the log files since the last full or incremental backup. To restore, only the last full backup and the last incremental are required.

### Differential Backup

The log volume or file is mirrored to an alternate location via a hardware or software mechanism.

1. The Exchange writer informs the requestor when it is ready for a snapshot to be taken.
2. The requestor has 10 seconds to complete the snapshot. This includes mirroring the remaining log file or blocks in the log file disk that are not in sync and severing the relationship.

The difference is that the differential backup does not truncate the log files generated since the last backup. This means that each differential backup that is taken contains only the log files since the last full or incremental backup. To restore, the last full backup and each of the subsequent differential backups are required.

### Copy Backup

A copy backup follows a similar process to the full backup. The difference is that the copy backup does not truncate the log files.

## Exchange Streaming Backup API

The Exchange Streaming Backup API has been in existence since Exchange Server 5.5. It has been slightly updated, but for the most part left alone to support the array of backup

applications that use the API. The Exchange Streaming Backup API's backups are referred to as streaming backups and are also an online backup process as they require the Exchange Information Store to be running. In Exchange Server 2007, the mature streaming API did not require any updates. To use the API, begin the backup process by starting the backup application. The backup application informs the ESE that it is entering a backup mode, and then a patch file (.pat) is generated for each database in the backup (assuming this is a full backup). During an online, full backup, the database is open for business, and transactions can be entered into the databases. If a transaction causes a split operation across the backup boundary (the location in the EDB file that designates what has and has not been backed up), the affected page before the boundary is recorded in the .pat file. A separate .pat file is used for each database that is backed up, such as Priv1.pat or Pub1.pat. These files are seen only during the backup and restore processes. During differential or incremental backups, a patch file is not created.

When the ESE enters a backup mode, a new log file opens. For example, if Edb.log is the current open log file, Edb.log closes and is renamed to the latest generation, and a new Edb.log is opened. This indicates the point when the ESE can truncate the logs, after the backup is complete.

When the backup begins, backup requests that ESE read the database and sequence the pages. After sequencing, the pages are grouped into 64-KB chunks (eight pages) and loaded into RAM. ESE then verifies the checksum on each individual page to ensure data integrity. If any page has a calculated checksum that does not match the checksum recorded in the page when the page was written to disk, the Information Store evaluates whether it is a single bit flip error (meaning the single bit is a 1 when it is supposed to be a 0 or vice versa). If the error is a single bit flip, an attempt to correct the error is made. If the error is not a single bit flip, the backup process stops and records an error message in the event logs. Backup does this to prevent the storage of damaged data.

The backup API also takes this opportunity to scrub the pages if the "zero out deleted pages" flag is set. It is only during an online streaming backup that this occurs and only after the original data has been moved to alternative storage that the streaming API overwrites pages that contain zero references from other pages (including indexes and mail items) with a set of alphanumeric characters. The nice thing about all this is that when you get a successful full, online backup of your Exchange databases using the Exchange agent from your software vendor, you can be certain that the database on your disk library or tape has complete integrity, because every page was read into RAM, its checksum calculated, and then copied to disk or tape. This is also different than a VSS backup which each database page's checksum is not verified. As an administrator, this is something to consider when planning a backup strategy.

Once the backup is successfully completed and all the pages are read, backup copies the logs and patch files to the backup set. The log files are then truncated or deleted at the point when the new generation started at the beginning of the backup. The backup set closes, the ESE enters normal mode, and the backup is complete.

In an incremental or differential backup, only the log files are affected. Operations that involve patch files, checksums, or reading pages sequentially are not executed.

To recap, here are the steps of the full backup process:

1. The backup starts, a synchronization point is fixed, and an empty patch file is created.
2. Edb.log is renamed to the next log number regardless of whether it is full, and a new Edb.log is created.
3. The backup for the current storage group begins.
4. A .pat file is created for each database that is being backed up in the storage group, and the database header is written into the .pat file.
5. During backup, split operations across the backup boundary are written into the .pat file.
6. During backup, Windows Server 2003 Backup Utility copies 64 KB of data at a time. Additional transactions are created and saved as normal. Each page's checksum is calculated and compared to the checksum recorded for that page in the page. The checksums are compared to ensure data integrity on each page.
7. If the database is configured to overwrite deleted pages, they are overwritten with an alphanumeric character set.
8. Logs used during the backup process (those from the checkpoint forward) and the patch files are copied to disk or tape.
9. If the database is configured to overwrite deleted pages, the overwritten pages are written to disk.
10. The old logs on the disk are deleted.
11. The old patch files on the disk are deleted.
12. Backup finishes.

## Types of Streaming Backups Supported

The streaming backup programs have a standard set of methods they can use to back up Exchange databases. The Windows Backup Utility takes advantage of all the features and can be used in a recovery strategy. Other streaming backup applications add feature sets

that are specific to each vendor. This section describes what happens during each of these events.

### Full Backups

1. The backup application starts and informs the ESE of the backup; a patch file is created for each database in the backup. A new log generation is opened to receive incoming database requests.
2. The backup application reads the database file into memory page by page. As the database is read into memory, the checksum is verified and the Zero Out Deleted Pages flag is checked.
3. If there are any uncorrectable -1018 or -1022 errors, the database stops on those pages, and the backup does not continue.
4. If there are any pages that are marked for deletion and the Zero Out Deleted Pages flag is enabled, these pages are overwritten with an alphanumeric set of characters and written back to disk.
5. Once the entire database passes through steps 1 through 4, the database, patch file, and logs are copied to the backup media, the logs are truncated to the log generation opened in step 1, and the database header is updated with a current timestamp under the backup complete heading.

### Incremental

1. The backup application starts and informs the ESE of the backup. A new log generation is opened to receive incoming database requests.
2. The backup application reads the log files on disk and copies them to the backup media.
3. Once all of the logs up to the log generation opened in step 1 are copied to media, the logs are truncated and the backup is complete.

### Differential

1. The backup application starts and informs the ESE of the backup. A new log generation is opened to receive incoming database requests.
2. The backup application reads the log files on disk and copies them to the backup media.
3. Once all of the logs up to the log generation opened in step 1 are copied to media, the backup is complete.

### Copy

A copy backup follows a similar process to the full backup. The difference is that the copy backup does not truncate the log files.

### Restore Process

Before you begin the restore process, you must dismount the databases to make them inaccessible to users. You can do this by using the Exchange Management Console (EMC) or the Exchange Management Shell (EMS).

When a restore operation begins, the store informs the ESE that a restore process is starting and ESE enters restore mode. The backup agent copies the database from the tape directly to the database target path. The associated log and patch files are copied to the server in a temporary location specified by you so that they aren't saved to the same location as current files in the production environment. If you happen to select the production path as your temporary path, you can overwrite log files and cause a logical corruption of the current production database. So, ensure that your temporary path is not your production path.

After the log and patch files are restored to the temporary location, a new restore storage group needs to be created for the purpose of restoring the database. The database is then copied from tape to the temporary location (and into the restore storage group). Then the patch file data and the log files from the tape backup are copied into the database by the restore database engine.

ESE processes the current logs, bringing you back to the point in time of your database backup (assuming you have all the transaction logs available from the last full, online, successful backup to the point of the disaster). After this is complete, ESE performs some cleanup by deleting log and patch files from the temporary location and deleting the restore storage instance. Then the storage group is mounted into the production environment, and your database is mounted, too.

## Other Exchange Server Components

In addition to the Exchange database, several other items must be included in backup planning. Some of these items are common among different roles and can be backed up and restored in a similar fashion. For example, a system state backup is useful to have on all server roles. Some of the items are unique per server role and are useful only on those specific server roles. Refer to Table 16-1 for additional server components to back up.

**Table 16-1 Additional Server Components to Back Up**

| Server role | Data to back up | Location/method |
|---|---|---|
| Mailbox | Mailbox or public folder database and logs | Streaming Backup or VSS Backup |
| | Content Index | No backup necessary, rebuild index during recovery |
| | System settings | System state backup |

**Table 16-1 Additional Server Components to Back Up (Continued)**

| Server role | Data to back up | Location/method |
|---|---|---|
| Hub Transport | System settings | System state backup |
| Client Access | Client Access configuration (IMAP settings, availability services, etc) | \ClientAccess*.* |
| | Exchange ActiveSync configuration | System state backup |
| | Web services configuration | |
| | Autodiscover services configuration | |
| | System settings | |
| Edge Transport | ADAM Customizations | Clone Config (ExportEdgeConfig.ps1) |
| | System settings | System state backup |

Additional information is available on Microsoft's Web site at *http://technet.microsoft.com/en-us/library/bb124780.aspx*.

# Backup and Restore Strategies

Your restore strategy determines your backup strategy. These operations cannot be planned separately. When you plan what backup strategy is best for your environment, first think about how and where the backups will be restored. This first step leads your planning down a path that will suit your environment and overall requirements better than thinking about backups first.

For example, how will you need the database backup to be available in the event of a restore? It could be in the form of a file on a tape that can be copied to a production location. It could be in the form of a disk that can be mirrored to a production location. It could be in the form of several backup files on a disk, or on several disks. The point is that consideration must be given to what types of restores are possible in most of the recovery scenarios that may arise.

You need to ensure you have enough hard drive space to restore both the database and the log files. If you generate 2000 log files in a single week, you have 2 GB of information to (potentially) restore. Add to that your database sizes, and you begin to see why you need to plan your restore strategy along with your backup strategy.

In addition to the backup and restore technology considerations, consideration of Service Level Agreements (SLAs) for backup times, restore times, and mail availability is important. These business level requirements are critical to consider when planning what methods to use for recovery. The SLAs need to meet the requirements of all consumers of the Exchange system. This includes individual users, applications, and business

processes. With this wide array of consumers in the Exchange environment, SLAs are difficult to agree upon, and the complexity of the system will increase. As a point of reference, look at the SLA scorecard shown in Table 16-2.

**Table 16-2 SLA Scorecard**

| SLA | Standard maximum time* | Premium maximum time** | Actual observed |
|---|---|---|---|
| Exchange Service availability | 99.999% | 99.999% | 99.99875% |
| | M-F 0700-1800 | S-S 0000-2400 | |
| Mobile Service availability | 99.999% | 99.999% | 99.997% |
| | M-F 0700-1800 | S-S 0000-2400 | |
| Application relay availability | 99.999% | 99.999% | 100% |
| | M-F 0700-1800 | S-S 0000-2400 | |
| Outlook client availability | 99.999% | 99.999% | 100% |
| | M-F 0700-1800 | S-S 0000-2400 | |
| Outlook Web Access availability | 99.999% | 99.999% | 99.98% |
| | Business Hours | S-S 0000-2400 | |
| Single mailbox restoration | 4 hours | 1 hour | 2 hours |
| Mailbox database restoration | 4 hours | 1 hour | 2 hours |
| Mailbox server restoration | 5 hours | 2 hours | 7 hours |
| Per mailbox e-discovery | 5 days | 2 days | 3 days |
| Mailbox item recovery | 1 week | 1 week | 5 days |

***Standard is defined as all users that pay the default chargeback rate per user**

****Premium is defined as all users that pay above the default chargeback rate for a higher level of service**

When evaluating the restoration and availability requirements, it becomes easy to draw the line in terms of the technical architecture to support those requirements. One critical area to look at is mailbox and database sizing. Notice that a single mailbox restoration and a mailbox database restoration are separate. This is normal and is due to the size of the mailbox and the number of users on a single mailbox database. There are several ways to leverage the Exchange configuration to support different SLAs. The most common is to leverage the size of mailboxes and number of users per database. These two components are what define the database size and the related backup and restore times per server. An example of this would be to look at the average restore time for an existing server. If we assume there are several servers that contain five databases (one per storage group) each, have 100 users per database, and each user has a mailbox limit of 400 MB. Each database would be about 47 GB. If the concern is the time it takes to restore the entire server, reducing the number of users per server or reducing the size of the mailbox

per user are good options. In either case, you need to perform an offline defragmentation to reduce the physical size of the database. On the other hand, if the number of servers was the concern, a consolidation could occur to bring all users onto a single server by growing the number of storage groups and databases for a single server. This would affect the time it would take to restore the entire server, yet the individual mailbox database restoration time would remain the same. This is an important concept to understand when planning the underlying plan for server sizing.

You can find additional information on storage planning at *http://technet.microsoft.com/en-us/library/c5a9c0ed-e43e-4bc7-99fe-7d1a9cb967f8.aspx*.

## Testing Restore Capabilities

Testing the restoration of backups on a regular basis solves three critical problems for organizations. The first problem it solves is keeping technical staff current with restoration techniques and skills that are normally only used during a disaster. The second problem it solves is fully validating the backup that was taken from the system. The third problem it solves is finding issues that exist in the backup process that otherwise would be found only during a real disaster.

All three of these problems are addressed with a single restore that occurs on a regular basis. It is recommended that a full restoration be tested at least once a quarter. This restoration can be done by different staff members each quarter, or a different service can be tested each quarter (such as testing a stand-alone mailbox restoration one quarter and a cluster failover the next). This way the restoration tests are not intrusive on other projects or daily operations yet remain an important component of regular activities.

The most common of these restores is the Mailbox server restore. This is the core component of Microsoft Exchange Server 2007 and will be the most important server to restore in the event of a site failure or a single server motherboard failure. For this reason, you will walk through a common scenario to test this type of restore.

There are two methods that are generally recommended to restore Exchange Server 2007; the first is to restore all databases of a failed server to a similar server, and the second is to restore the databases from the failed server across the remaining Mailbox servers in the environment. The first option is similar to the restoration technique in earlier versions of Exchange. The second option is new to Exchange Server 2007. The first option is reviewed; however, the same techniques are applicable to the second option as well.

It is important to understand a feature in Exchange Server 2007 referred to as Database Portability. This feature provides the ability of databases in Exchange Server

2007 to be portable between Exchange servers in the same Exchange organization. What this means is that an Exchange database that resides in Storage Group 1 named DB1 on a server named Exch1 can be shut down (or otherwise made consistent), moved to Storage Group 2 on a server named Exch2, and mounted with no further actions or modifications. To make this work, you must first create a placeholder database in Storage Group 2 on the Exch2 server and then enable the This Database Can Be Overwritten By Restore option.

To test a full server restore and validate the mailbox contents, a new Active Directory forest should be installed on a spare server in a lab environment. This server can be a virtual server, workstation-class computer, or other underpowered computer that is available. With the new Active Directory forest, a new Exchange server must be installed. This Exchange Server can be installed into an Exchange Organization with the same or a different name. The Exchange server itself can be installed with any name you choose, as well. With a new server installed and configured similarly to the production server, the storage groups and placeholder databases must be created. At this point, the databases from the production server must be restored by whatever means your backup strategy allows. Remember, it is important to follow real restoration procedures here. After the databases are restored, they should be mounted, and mailboxes can be connected to the databases to validate the contents of the mailboxes. Should you perform these actions in a production environment instead of a lab environment, ensure that test mailboxes are connected to the restored mailboxes and not the user mailboxes.

During a real disaster, or in a test Exchange organization that is left available, it is easy to repoint the users' Active Directory settings to the restored databases. Simply run the following command from the Exchange Management Shell:

```
move-mailbox -configurationonly -targetdatabase <new_database_name>
```

You can't perform a restore without knowing that your backups are working. Verify your backup jobs complete every day and test a restore quarterly. Failure to verify backups is a common mistake because it is easy to assume that backup tapes are swapped and that data is backed up properly. Make it part of your regular routine to review all backup logs and perform a restore to ensure that restoration and recovery is feasible.

---

**Note** Verifying backups does not have to be difficult; it can be monitored through an automatic system like Microsoft Systems Center Operations Manager or through custom scripts that parse the event logs looking for backup success events. It is important to look for the backup success events instead of just the backup failure events; this is so that you know the backup did not run too long without reporting a failure.

Testing restoration procedures of the features other then the mailbox databases and Mailbox servers is also important. In many cases, this will be a rebuild operation. In other cases, this will be an intricate process. For more information on these procedures, refer to *http://technet.microsoft.com/en-us/library/aa998890.aspx*.

## Recovering an Exchange Mailbox Server

Planning for recovery of an entire Exchange Mailbox server can be a lengthy process. It is the core server in the Exchange infrastructure. It is the reason all of the other components exist. Consider the following three things when planning the type of recovery strategy you will use:

1. **The Active Directory location of the restore.** Is the Active Directory site intact, and are all Active Directory objects that were previously there intact?
2. **The supporting Exchange Server infrastructure.** Are the Client Access server and Hub Transport server roles in existence in the site where the Mailbox server is being recovered? If there are Unified Messaging, Edge, SharePoint, or Rights Management services in use, will the recovered Mailbox server be able to interact with them?
3. **The server that is being restored.** Is the server a clustered Mailbox server, a stand-alone server, or a stand-alone server that is functioning as a single-node cluster?

The most common recovery strategies for the full Mailbox server are full server restores and Exchange application rebuilds. These two strategies are typically used with warm (also known as stand-by) servers, and using dial-tone servers is the first step.

### Full Server Restores

Full server restores occur when a backup of the Exchange databases is maintained somewhere in addition to a backup of the Windows Server data. These are not commonly used with Exchange installations because Exchange does not rely heavily on the Windows Server system state during its operation. The steps involved in this restore process are as follows:

1. Obtain replacement hardware with components that match the original server's hardware components. This can include RAID controllers, network interface cards, and so on.
2. Rebuild Windows Server to the same version, service pack, and driver versions as the original server.
3. Restore the Windows Server system state and file system backup.
4. Re-install Exchange Server and service packs.
5. Restore the Exchange Server 2007 databases.

At the end of this process, users can access their mailboxes with no reconfiguration. No database recovery is needed, and the server is expected to last as long as the components allow. The Active Directory and Exchange considerations previously discussed need to be considered in this strategy. One exception to this is where Exchange Server 2007 resides on the Active Directory server and is combined with the Hub and CAS roles. In this case, all components are restored during the steps outlined.

This is a reliable solution; however, it is costly and carries a high amount of administrative overhead. Ensuring the exact same components can be obtained usually requires keeping them on hand as hardware components have a shorter cycle then software components. The time it takes to rebuild and restore a Windows Server 2003 computer is more than the time it takes to only rebuild the server and reinstall the applications. These limitations make this option a difficult choice in situations in which e-mail is in the least bit critical.

The solution is limited by the Recovery Point Objective (RPO) and the Recovery Time Objective (RTO) it supports. The RPO can be increased with third-party technologies that replicate the database to the recovery site. The most common technologies are VSS copies/log shipping the data over the network or replicating the changed tracks on the disks. The RTO is very high; this is first due to the large amount of work that must be done to prepare the environment and is secondly due to the large amount of time required to restore the data from backups.

## Exchange Application Rebuilds

Exchange application rebuilds is a broad term. It refers to a rebuild of the Exchange Server application followed by a recovery or restoration of the data. To rebuild the Exchange application on a server, the application can simply be installed from scratch on a new Windows server, or the application can be installed on a stand-by server using the `/m:RecoverServer` switch or the `/RecoverCMS` switch. The first thing an administrator needs to know before walking through the steps is what type of Exchange server will be reinstalled. If it is a clustered server, use the `/RecoverCMS` switch; if it is not clustered, use the `/m:RecoverServer` switch.

To use the `/m:RecoverServer` switch, complete the following steps:

1. Using Active Directory Users and Computers, reset the computer account for the server name that you are attempting to recover.
2. Verify all server names, volume configuration, and directory paths are the same as the previous server.
3. From a command prompt, change into the Exchange Server source directory and run the following command:

```
Setup.exe /m:RecoverServer
```

At this point, the Exchange Setup program queries Active Directory to obtain the configuration information for the server name you are attempting to recover. The server roles previously installed and the locations of the data are used to configure how Exchange Server is installed. Users should not need to change any information, and administrators should not have to move user configuration.

To use the `/RecoverCMS` option, complete the following steps:

1. Using Active Directory Users and Computers, reset the computer account for the server name that you are attempting to recover.
2. Verify all server names, volume configuration, and directory paths are the same as the previous server.
3. Build the cluster server with the same configuration as the previous cluster. Single copy clusters cannot be restored to clustered continuous replication clusters and vice versa.
4. Install the passive Exchange Server Mailbox server role on the node of the cluster you are recovering to.
5. From a command prompt, change into the Exchange Server source directory and run:

   ```
   Setup.exe /RecoverCMS /CMSName:<CMSName> CMSIPAddress:<CMSIPAddress>
   ```

6. If you are recovering an SCC cluster with several Exchange Clustered Mailbox Servers (CMSs), perform steps 4 and 5 until all Exchange CMSs are installed on active nodes.
7. Recover Exchange Mailbox databases. This can be from backup, from a disk-based replication, or from a log shipping–based replication.
8. Install one or more passive nodes in the cluster.

At this point, the Exchange CMS is recovered and operational. Users should not need to change any information, and administrators should not have to move user configuration.

For both Exchange recovery strategies, the Active Directory and Exchange considerations need to be considered. No other Exchange roles can reside on a clustered Exchange server, which introduces a requirement for these to already exist in the site. These should have already been rebuilt, already exist, or be the next item on the list to rebuild.

These are reliable solutions; the cost and administrative overhead can vary depending on what configuration is used. The servers used do not have to match exactly, although for SCC, the servers do have to be on the Windows Hardware Quality Labs (WHQL). The Exchange CMS name must match the CMS name that was previously used; however, the

server names, IP addresses, and so on, do not have to match. The time it takes to use a recovery option is much less then the time it would take to manually re-create the configuration and validate that the configuration was entered correctly.

The solution is limited by the RPO/RTO it supports. The RPO can be increased with third-party technologies that replicate the database to the recovery site. The most common technologies are VSS copies/log shipping the data over the network or replicating the changed tracks on the disks. The RTO is still higher than a high availability strategy as some amount of work still needs to be done. The RTO is lower than a restore strategy as more of the data that already exists and is necessary for the application to run is used during the process.

## Dial-Tone Servers

Dial-tone servers are similar to dial-tone databases and are simply empty configurations that are used in an effort to restore service to end users. Dial-tone servers or databases do not meet any of the requirements of a clustered, stand-by, or highly available service. This is because the messaging data is not readily available during a dial-tone recovery. Basic e-mail functionality is enabled immediately; however, e-mail data, rules, calendar data, UM settings, mobile device configuration, and so on, are not recovered. The advantage to this solution is that users are able to operate in a limited form until the rest of the data can be recovered or re-created.

To implement a dial-tone recovery, complete the following steps:

1. Fully install an Exchange Mailbox server and create all of the necessary storage groups, databases, and other environmental configurations.
2. If this is a database dial tone, create storage groups and databases on existing servers that are managerially designated for use in a dial-tone recovery strategy.
3. If dial-tone recovery is needed—for example, if the server or site suffered a failure—use the `Move-Mailbox ÐConfigurationOnly` cmdlet and point the all mailboxes that must be moved to the dial-tone databases.
4. If the messaging data becomes available, it can either be recovered to the dial-tone server and imported to the mailboxes, or it can be prepared on a replacement Exchange server and the users can be moved to the replacement hardware.

This solution has a very low RTO for the service and a low RPO; however, the RTO for the data is usually higher. This has some advantages for organizations that require e-mail services to process business functions but do not require historical data to process those same business functions.

## Recovering an Exchange Mailbox Database

Exchange Server 2007 mailbox database recovery is usually more complex than recovering the Mailbox server itself. Mailbox database recovery can involve simple tools such as the Exchange Disaster Recover Analyzer, or it can involve complex tools such as the Exchange ESE Utility or the Exchange Information Store Integrity tool.

There are times when a clean backup of a mailbox database cannot be obtained, and some form of database recovery must be performed. For the most part, Exchange Server 2007 has simplified the recovery process by enabling database portability. This is a feature built into the database that allows a database to move to and be mounted on any server regardless of the server name. This is important to recovery strategies as once a database is in a consistent state and mountable, it can be placed almost anywhere to be used.

The trick is getting the database into a consistent state. There are several ways to accomplish this task. If you perform a restore of your database and find that it is corrupted or unmountable, a good first step is to go into the Exchange Disaster Recovery Analyzer in the toolbox and point it at the database. This wizard walks you through finding the database and running Eseutil.exe in recovery mode in an attempt to make the database mountable.

If this is unsuccessful, two options exist. The first is to manually run a soft recovery on the database with Eseutil.exe by running the `Eseutil.exe /r` switch against the database. The second is to run an `Isinteg.exe Ðfix` to fix any errors that are in the database pages from an application perspective.

## Recovering a Single Exchange Mailbox

Events have been covered where your site, server, and database have all failed. In these scenarios, entire servers have been restored to working order, and mailbox databases have been recovered. However, the most common recovery situation Exchange administrators experience is mailbox recovery. It is more common that a mistake occurs against a single mailbox level than on an entire database or server level. Because of this, the ability to recover a single mailbox is a crucial part of your overall planning.

Recovering a single mailbox in Exchange Server 2007 is relatively simple; in fact, once you have the database in an RSG waiting for the restore, it is as simple as running the following command:

```
Restore-mailbox Ðidentity <DisplayName> -RSGDatabase <RSG\MailboxDatabase>
```

This command is the simplest method of recovering a single mailbox database and restores the mailbox data to the mailbox associated with the <DisplayName> specified. This is good and it is useful. In fact, it is probably the most common scenario for deleted mailbox recovery; however, there are other options to accomplish this task.

If you are in a situation in which an employee has left the company and the previous manager wants to see the state of the mailbox prior to departure, a restore of an old mailbox database backup would be restored to the RSG. You would then need to put the employee's mailbox into the manager's mailbox. You would run the following command:

```
Restore-Mailbox -RSGMailbox 'Ex-Employee Name' -
RSGDatabase 'RSG\Mailbox Database' -id 'Manager Mailbox' -TargetFolder 'Ex-
Employee's OldEmail'
```

It is important to build in the capability to restore a single database to an RSG to support this capability. Without it, restoring a single mailbox will be available only with third-party tools.

### Real World Planning a Deleted Item Retention Strategy

All of these recovery strategies are great; however, it is well known amongst seasoned administrators that e-mail has a tendency to be accidently deleted. This presents a problem for e-mail users and administrators. The need to quickly recover e-mail that has been deleted is usually of the utmost importance and without proper planning can be cause for a stressful situation. I have been in this position several times. A senior executive was working late on a proposal or project and, of course, I was working late updating server patches. Suddenly an Instant Message pops up from the senior executive asking for a way to recover one of her e-mail items. I dropped everything I was doing to respond to the request (after all it was late, and this could make a good impression). After some brief discussion over IM, I discover that during the course of the evening the executive had been referencing several e-mails from her employees to generate a report for her boss. To use the e-mails, the executive was opening and closing each message and then closing it with a familiar keystroke. When she completed her work, she printed and reviewed her report only to find an error in some of her data. She went back to her inbox to find the e-mail that contained the correction only to discover the e-mail was gone. She searched the folder and the deleted items folder to no avail.

After some testing on my own client, I found that the keystroke she was using was common for a separate application; however, it caused our corporate e-mail client to permanently delete the e-mail item. This is when I was glad that our deleted item retention policy was still using the default 14 days. I could confidently respond that I could recover any of the items that she needed to complete her work for the evening without the process of database restores and item restores from the Exchange Management Shell.

It is important to plan an appropriate deleted item retention strategy and ensure that the amount of time specified can be stored within the database space planned for. A general rule of thumb is 10 percent of the database for the default 14-day deleted item retention. Of course, should you go over the 14-day deleted item retention and need to recover specific items from a database restore, the `Restore-Mailbox` command does the trick. To use the command, simply restore the database to an RSG then run the cmdlet and specify which messages you want to restore to which mailbox in production.

## Backing up an Exchange Mailbox Server

Now that you have an understanding of what is required to recover Exchange servers, there is enough context to talk about how to back up Exchange servers. There are several ways to implement the two technologies discussed previously to accommodate the recovery strategies. These include streaming or VSS-based full, differential, incremental, and copy backups in conjunction with ensuring certain information is available in Active Directory or in a transportable copy of the server.

For server backups, ensure the system state, the registry, and the applications that support the server installation are included in the backup definition. These backups do not need to be taken every night; however, they should be taken before and after all patching and software upgrade processes.

For server rebuilds, make sure you keep thorough documentation. This could be in the form of meticulously written configuration documentation, a standard automated installation for Exchange servers, or by using a third-party configuration management system. In any of these cases, the backup should be simple and it should be tested every time a new server is built.

For server recovery, there are more options for Exchange Server. In the case of using the information in Active Directory, the operating system needs to be rebuilt to a state that supports Exchange, but not to the same state it was in previously. This removes the need to maintain a backup or copy of the system. You can also recover from a storage area network (SAN). In this scenario, logical unit numbers (LUNs) are maintained on the storage area network storage array and can be reused in the event of a physical server failure. The server itself must be rebuilt to match the previous configuration (including HBAs, drivers, SAN connections, and so on) as in the first scenario; however, the data is not restored from a backup. To recover, the rebuilt server is simply plugged into the SAN, given access to the LUNs, and powered on. This scenario also gives you the flexibility of replicating operating system boot LUNs to other storage arrays or storage array volumes.

## Backing up an Exchange Mailbox Database

Backing up an Exchange mailbox database can be difficult to plan. There are many things using the database throughout the day that should not be interrupted for backups. User access, content index rebuilds, and online maintenance are the most common things using the database during the day. It is best to schedule the backup operations to take place during a time that will not conflict with these things and still complete every night.

The following two methods are commonly used; they are well-tested methods that meet certain recovery point and recovery time objectives:

- A weekly full backup plus a daily incremental backup
- A daily full backup

---

**Note** As an alternative to these technologies, some third-party vendors enable non-VSS and non-streaming backups through custom disk mirroring and/or file system drivers. A careful evaluation of these third-party technologies is necessary to ensure supportability and technical soundness prior to any production implementation. Often Microsoft will not support these technologies, leaving the first level support to third-party vendors.

### Using a Combination of Weekly and Daily Backup Methods

The first method is a weekly full backup plus a daily incremental backup. This method can be done using a streaming-based backup or a VSS-based backup. This method generally keeps backup times lower due to the speed of the incremental backups. With a lower backup window, databases can be larger, which in turn allows for higher mailbox sizes while still allowing user consolidation. This is important in many environments, which has made this a popular method. The lower backup window also does not bump into the critical online maintenance, which should complete once per week and should not bump into index rebuilds or general user activity. All of these attributes are good for backups; however, there are downsides to this method.

The most obvious downside is the number of backups that need to be restored to accomplish a full recovery. Another downside is that the potential corruption of any one of the incremental backups could be detrimental to the overall backup strategy; however, using only full backups would allow a recovery of any one of the latest full backups.

### Using a Daily Full Backup Method

The second method is a daily full backup. This method can be done using a streaming-based backup or a VSS-based backup. This method generally has longer backup times every night due to the amount of data that is backed up. Thankfully, disk mirroring VSS

backups are able to drastically reduce the time and impact to the hosts. If you are unable to use one of these solutions, a software VSS provider or a streaming backup program will suffice. Regardless of the technology in use, a longer backup window should be planned for to ensure that a successful completion of the backup is obtainable. Often issues such as long-running consistency checks, media mounting errors, and midbackup failures are not planned for and backup windows are overrun.

During the planning stages of a VSS-based full backup solution, there are several things you need to understand:

- The time it takes to complete a consistency check
- The time it takes to synchronize the VSS media with production spindles
- The behavior of media mounting failures, including disk and tape media
- The behavior of midbackup failures, including the ending state of the disks

During the planning stages of a streaming-based full backup solution, there are several things you need to understand, as well:

- The impact to the host if the Zero Out Deleted Pages flag is enabled
- The time it takes to perform a full backup
- The behavior of media mounting failures, including disk and tape media
- The behavior of midbackup failures, including the ending state of the disks

These are all important to understand and plan for to ensure a smooth operational state of your backups. Far too often, these are overlooked and managing backup operations becomes a long-running task that is never resolved. This can result in backups that are incomplete or nonrestorable in the event of a failure.

## Backing up a Single Exchange Mailbox

Backups of single Exchange mailboxes is the simplest topic discussed in this chapter. To put it simply, Microsoft does not natively support this feature. The two backup technologies, streaming and VSS, allow for only full database backups and restores. However; this does not mean that single mailbox or brick-level backups are not possible. In Exchange Server, it is feasible that a third party could create a backup product that would back up each mailbox individually. This is usually done through the MAPI interface in the same manner that an Outlook client logs into a mailbox and is able to read all of the items in a mailbox. This has been accomplished in previous versions, and vendors have added and dropped this capability frequently. Some vendors have implemented this strategy to the level of backing up at an item level, so that individual messages can be restored from an actual backup file.

These third-party applications can add a lot of value to your environment, but performance and timelines of these solutions introduce some serious problems. If you are looking at putting one of these solutions into place, seriously consider moving that backup from the production copy of the database to a replica copy of the database. That replica could be a CCR, LCR, or other replication technology replica, as long as it is not taking up time on the production volume.

An alternative to using a brick-level backup solution would be to go with an application that allows for a hands-off backup/restore strategy of a mailbox using a full database backup/restore through the recovery storage group. This strategy would not impact performance on the production system to the same degree a brick-level backup solution to the same degree as a brick-level backup solution would.

## Planning for Corruption

Corruption is a fact of life. At some point in time, a database under your control will become corrupted. This is something that you should plan for and be well prepared for, with knowledge of the tools, procedures, and calmness required to deal with this critical situation.

You should be familiar with the following tools:

- Eseutil
- Isinteg
- MfcMAPI

Familiarity with these tools allows you to manage the ESE database engine (eseutil), find and fix errors in the Information Store layer (isinteg), and look into specific mailboxes through the MAPI protocol (mfcMAPI). It is important to know the basic difference between the three components so that you understand where a particular problem might lie. In Figure 16-3, the database instances (ESE) are at the bottom; this is where the data is actually stored. The Information Store sits on top of the databases and is a single process that manages access to all of the individual databases; this process controls how all information gets into and out of the databases. The MAPI interface connects to the Information Store; this controls how Outlook clients view and communicate with the Information Store and ultimately the database. Outlook uses the MAPI interface to communicate with Exchange. You see that each tool can affect a separate portion of the stack: Eseutil.exe can interact with the ESE database directly; Isinteg.exe can interact with the databases in the context of the Information Store; and MfcMAPI.exe can interact with the Information Store through the MAPI interface in the same way that Outlook can (however, it can do this without any restrictions).

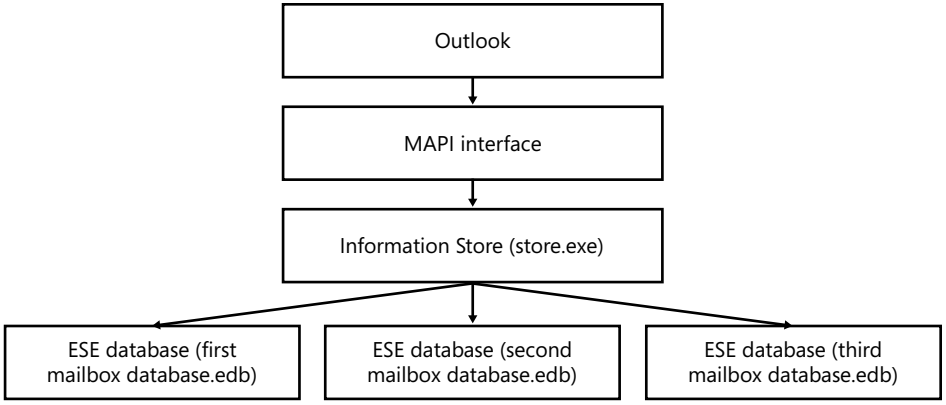



**Figure 16-3** Layers between Outlook and the database

## Implementing Backup Strategies

The three major requirements that need to be considered when implementing any of the previous strategies are RPO/RTO, financial, and environmental. These requirements define how the strategies are implemented and how they operate.

RPO and RTO are the most commonly discussed of these requirements. The RPO is the Recovery Point Objective and is the desired point in time the system is recovered to. This means that if you back up the system once a week on Wednesday at 10 a.m., your recovery point objective for the rest of the week is Wednesday at 10 a.m. as that is the only point in time that you can recover to. The RTO is the Recovery Time Objective and is the desired amount of time it takes to recover the system. This means that if your backup methodology allows you to plug in a USB cable, click a button, and wait 10 minutes to have the system fully recovered, your backup time is about 11 minutes (assuming the cable and button work take about one minute to complete). Combining these two is common. Usually systems with high RPOs come with high RTOs, and the same is true for the reverse. The problem is that the financial cost of these usually increases and decreases together as well: low RPO/RTO scenarios usually cost more, and high RPO/RTO scenarios usually cost less.

This leads to the second of the major requirements: financial cost. Financial cost is an important consideration and is likely one that heavily influences what strategies are used. For example, a fully geographically dispersed cluster with replicated storage arrays has the lowest RPO and RTO for e-mail database environments and is usually an expensive solution. On the other hand, an encrypted backup tape mailed to the recovery site where it is restored to new servers has a very high RPO and RTO and is about as inexpensive as you can get.

The environmental consideration is the last major requirement and one that requires significant planning. This consideration includes what Exchange Server roles are on the server that needs to be recovered, to what Active Directory site the server is recovered, and what infrastructure is in place between those two sites. If the server that needs to be recovered contains only the Mailbox server role, then a clustering solution may be an option; however, the Client Access server and Hub Transport server must already reside in the Active Directory site where the Mailbox server is recovered to. If the only network connection between the two sites is a high-latency T-1, disk replication or cluster continuous replication may not be a feasible solution to get the mailbox database copies to the recovery site.

The next section presents some common scenarios in which the previous requirements have been considered in devising strategies for services and data recoverability. These scenarios meet specific requirements and are for illustration purposes only.

## Clustered Continuous Replication with VSS

The first scenario is Clustered Continuous Replication (CCR) with Volume Shadow Copy Service (VSS) backups. This scenario combines the new CCR technology with the standard VSS framework. Using CCR and VSS in your environment provides several advantages to other data replication solutions; cost, supportability, and the restart rate are the primary advantages. Combining these into a scenario means your environment consists of the following hardware:

- One Microsoft Cluster Service cluster consisting of two Exchange Server 2007 Mailbox servers and one quorum device.
  - The Exchange Server 2007 Mailbox servers do not have to have matching hardware, although you should plan to run your production workload from either system at any given time.
  - The quorum device can be any supported quorum device in a Microsoft cluster. This includes a quorum disk, a quorum server in a Majority Node Set (MNS) cluster or a file share witness in an MNS cluster.
- One server to perform backups. This server can also act as the file share witness host for your MNS cluster. This server's hardware can be 32-bit or 64-bit and does not have to match the other server configurations. The VSS Requestor resides on the server performing backups. If the server uses a shared storage array, disk replicas are mounted on the backup server in order to perform a checksum verification.

This scenario is usually set up so that the cluster can operate on either node and can be failed over to either node for server maintenance or emergencies (planned or unplanned). The backup server can talk to both nodes over the network and has the

ability to talk to the VSS provider that is used. The schedule of backups depends primarily on the amount of data that is on each CCR group and the allowed backup window in the organization. The most common is a scenario in which full backups are taken on the weekends during an extended backup window and differential backups are taken during the week during a shorter backup window.

Two critical things to consider when implementing a scenario like this are

- What is the amount of data loss your environment can sustain (RPO)?
- What is the amount of time your environment needs to be restarted within (RTO)?

In the event of a planned failover, CCR ensures the last log file is closed and pulled over to the passive node in the CCR cluster. Once the log file is pulled over, the passive node can restart the clustered Mailbox server and the underlying database. On restart, the database should already be consistent due to the log replay mechanism on the remote side. This allows the restart to happen almost instantaneously and begin to serve client requests immediately after the failover is initiated.

In the event of an emergency failover, the process is similar. The passive node attempts to copy any log files from the active node that are closed or have not yet been closed. Once copied over, the passive node immediately attempts a replay of those log files to get any remaining data into the passive database. At this point the lost log resiliency feature kicks in to ensure the logs that were copied contain complete data. If the data is incomplete, the logs are not played into the database; instead, a new log generation is created, and any incoming data is processed through the new log generation while the incomplete data is disregarded. This allows the clustered Mailbox server to come online quickly and recall any data from the Hub Transport server's dumpster that may be available.

In both planned and unplanned scenarios, data loss is a possibility. If the last log cannot be copied to the passive node for any reason and the data cannot be resent through the transport dumpster mechanism for any reason, an amount of data equal to the number of messages that were partially available in those logs will be lost. This means the RPO for the failover portion can be variable; however, it is complemented in the scenario with a VSS backup, which provides a high-speed backup/restore mechanism to enable a more recent point-in-time copy on the VSS backup. The RTO is still relatively low for the failover and is also relatively low for the VSS restore (provided the VSS vendor in use allows for a fast restore).

Some of the drawbacks to this scenario are as follows:

- Failovers can cause data loss.
- If backups are taken from the replica, they are not easily moved between the production and replica nodes.
- Extending the solution between disparate sites can be difficult.

The data-loss component is hard to avoid with any scenario where the production copy of the database is not shared with, or synchronously replicated to, the passive copy. The amount of data loss is configurable through transport dumpster settings and investments in network infrastructure; however, the risk of data loss cannot be entirely removed. Additionally, if the VSS backup application takes backups from the replication writer, those backups can be restored only to the production writer. This means that a backup taken on the replica cannot be restored to the replica unless the Clustered Mailbox Server is failed over to the replica prior to performing the restore. In the event the backup is taken from the replica and the site where the passive cluster node becomes unavailable, there would not be a backup to restore.

## Single Copy Clustering with Streaming Backups

The second scenario is single copy clustering (SCC) with streaming backups. This scenario combines the traditional SCC technology with the legacy streaming backup API. Using SCC and streaming backups in your environment provides several advantages to the data replication solutions: minimal data loss, highest RTO, and minimal chances of database corruption. Combining these into a scenario means your environment would consist of the following hardware:

- One Microsoft Cluster Service cluster consisting of two or more Exchange Server 2007 Mailbox servers and one quorum drive.
  - The servers in the cluster should all contain the same components and be on the cluster WHQL.
  - The quorum drive must be accessible by all of the servers in the cluster.
- One server to perform backups. This server is responsible only for controlling the backups. The Mailbox servers themselves do most of the work to move the data in the database to an external location.

Two critical things to consider when implementing a scenario like this are:

- What is the amount of data loss your environment can sustain (RPO)?
- What is the amount of time your environment needs to be restarted within (RTO)?

In the event of a planned failover, SCC performs a restart of the clustered Mailbox server on the preferred passive node of the cluster. The active node shuts down the database of the clustered Mailbox server and starts the database on the passive node. Once all services start and the databases are mounted, the clustered Mailbox server is again ready for use.

In the event of an unplanned failover, SCC first attempts to restart on the active node hosting the clustered Mailbox server. If there are any limiting factors, like the active node's motherboard failing, SCC attempts to restart the clustered Mailbox server on the preferred passive node with the database in whatever state it is in. This means that if there

are any logs that have not been committed, any complete transactions are replayed into the database and any partial transactions are rolled back.

In both planned and unplanned failovers, this process ensures there is no data loss, which puts the RPO at zero. It also ensures the restart time is similar to that of restarting the services on the same node, which puts RTO at near zero.

The last component is the streaming backups. With streaming backups, the restoration of the data can take longer than the amount of time required for a VSS-based restore. However, because this scenario has such a highly available service, restoration for recovery purposes can be removed from the procedures. Doing this enables a lower-cost backup solution to coincide with a higher-cost availability solution.

Some of the drawbacks to this scenario are: server hardware costs, backup times are long, and database restores are lengthy. The hardware costs are unavoidable with a single copy cluster; it is the financial price that is paid for a high RPO and RTO. The length of the backup and restore times are also inherent in the solution. An alternative to this would be to make an environmental and financial investment in a VSS-based solution.

## Single Multi-Role Mailbox Server with VSS

The third scenario is a single multi-role Mailbox server with VSS backups. This scenario combines several roles onto a single server to provide a consolidated entity to manage in a recovery scenario. The roles that typically exist on a consolidated server are the Mailbox role, the Hub Transport role, and the Client Access server role. Combining this consolidated entity with VSS backups reduces the overall RTO and potentially reduces the RPO. This consolidation of roles means that the environment consists of the following hardware:

- One Microsoft Windows Server 2003 computer running two or more Exchange Server 2007 roles.
- One server to perform backups. This server is responsible only for controlling the backups.

This scenario is usually set up so recovery of the server to a hot, warm, or cold standby server in an alternate site is possible. The primary method of recovering the server is to perform a VSS restore to the same or to a new physical server. Consider these two critical things when implementing a scenario like this:

- What is the amount of data loss your environment can sustain (RPO)?
- What is the amount of time your environment needs to be recovered within (RTO)?

The unique thing about this scenario is that planned failovers are not possible; any downtime results in a service outage. This is also an indication of the amount of time required to recover the roles to a new server during an unplanned failover. The procedure involves

bringing up a new server as described earlier in the section "Recovering an Exchange Mailbox Server."

In both planned and unplanned recoveries of the system, data loss is a real possibility if the database is not shut down properly prior to the recovery.

The drawbacks of this scenario should be more obvious then the first two; there is no automatic failover of the Exchange server roles, and the recovery time can be lengthy depending on the state of the standby server.

### Review of Sample Scenarios

The three scenarios presented here are certainly not the only configurations. They are illustrated to show you the breadth of solutions available and help to guide you to what configuration items will affect the three major requirements in your environment.

The first scenario uses a lower-cost, high-availability solution with a higher-cost backup/restore solution. This enables a customer to maintain a reliable failover mechanism that allows a minimal amount of data loss accompanied by a fast backup/restore mechanism to maintain more frequent point-in-time copies that can be quickly restored if necessary.

The second scenario uses a lower-cost backup/restore mechanism with a higher-cost, high-availability solution. This allows a customer to suffer zero to minimal data loss and downtime while only allowing long backup and restore times.

The third scenario does not use a high-availability solution, yet uses a fast backup/restore mechanism to allow for a quick recovery in the event of a failure. This solution is more common among small and medium enterprises as well as large enterprises that are using some other form of high availability.

---

**Note** A common extension to the third scenario is to use LCR in conjunction with the multi-role Exchange server. LCR is implemented at the storage group level, so it can be configured more granularly, which is important to organizations that are on a tight budget.

## Operational Best Practices

To successfully back up or restore an operation, follow some best practices:

- Document your backup and restore procedures.
- Ensure copies of the backups are stored in an alternate location.

- Check your backup monitoring and logging system every day to ensure the Exchange server backups are successful from the previous night.
- Perform a trial backup and restore on a monthly or quarterly basis to ensure your solution is working and to keep your restore skills up to snuff.
- De-duplication technologies will benefit your backup media space consumption.
- If you use tape storage for your backups:
  - Routinely clean the tape drives according to manufacturer specifications.
  - Do not overuse tapes. Discard them after they reach the maximum number of cycles specified by the manufacturer.
  - Ensure that the raw storage capacity of your tape exceeds the compressed storage capacity of your database by a comfortable safety margin. If it does not, plan for tape changes when doing backups.
- If you use disk storage for your backups:
  - Routinely verify data integrity on the disk.
  - Ensure the raw storage capacity of your backup LUNs exceeds the storage capacity of your database by a comfortable safety margin. If it does not, plan for future growth.

## Summary

This chapter covered a lot of ground pertaining to backup and restore operations. It outlined how to perform restores of your Exchange databases, the general steps to follow to recover an entire server, and a brief overview of how the VSS feature in Windows Server 2003 can be used to keep your restore times to a minimum. If your databases become corrupted or something goes awry, be sure to use the techniques presented in this chapter to recover your databases and restore your Exchange information.