

Microsoft®
Exchange 2000
Server

**Best Practices for Deploying Full-Text
Indexing**

White Paper

Published: January 2002

Table of Contents

Introduction	2
Full-Text Searching Overview	2
What Data Is Indexed?	2
How Outlook Users Access Full-Text Searching	3
Full-Text Search Behaviors	3
Preparing Your Exchange Environment	4
Preparing Your Server	4
Preparing Your Exchange 2000 Organization	7
Deploying Full-Text Indexing	8
Creating a Full-Text Index	8
Optimizing Full-Text Indexing	8
Performing a Full Population	13
Setting a Schedule for Incremental Populations	14
Enabling Full-Text Indexing Queries	14
Notifying and Educating Users	14
Managing Full-Text Indexing	15
Checking the Size of the Index	15
Adding Users to an Indexed Server	15
Deciding When a New Full Population Is Required	15
Monitoring Full-Text Indexing	15
Objects to Monitor in Performance Logs and Alerts	16
Population-Related Counters	16
Query-Related Counters	17
General Performance-Related Counters	17
Troubleshooting	19
Event Viewer Errors	19
Population Process Is Slow	20
Population Process Found in a Paused State	21
Deleted Message Still Visible in Results	21
Wrong Location Is Displayed After Moving Index	21
Gather Log Entries	21
Language Settings Problems	22
Queries Failing During Initialization	24
Missing Performance Counters	24
Disk Bottlenecks	24
High Paging	25

Additional Resources.....	25
Appendix A. Sample E-Mail Message to Users	26

Best Practices for Deploying Full-Text Indexing

White Paper

Published: January 2002

For the latest information, please see <http://www.microsoft.com/exchange>.

Introduction

Microsoft® Exchange 2000 Server can create and manage full-text indexes for public folder and mailbox stores, providing users with fast searches. Earlier versions of Exchange searched every message in every folder and, as a result, search times increased as databases expanded. With full-text indexing, every word in a database is indexed, making faster searching possible.

This article contains best practices for deploying full-text indexing with Exchange 2000 Server.

Full-Text Searching Overview

Full-text indexing is a powerful search tool that functions differently from other types of searches, such as character-based searches. With full-text indexing, you create an index for selected mailboxes and public folders and then you make the index available for users to search. Full-text indexing searches are faster than the character-based searches because users are searching through the index instead of searching through the e-mail messages themselves. Full-text indexing searches are also more comprehensive than character-based searches because attachments are indexed in addition to e-mail messages.

An important part of your deployment plan is understanding how full-text searches function and educating users so they will get the search results they expect.

What Data Is Indexed?

When you deploy full-text indexing, you select the individual public folder or mailbox store to be indexed. Users can then conduct full-text searches on the messages and attachments contained in the public folder or mailbox store. By default, the index contains the subject and body of a message, along with names of the sender and recipient and any names that appear in the **Cc** and **Bcc** fields. The index also includes text from the following types of attachments: .doc, .xls, .ppt, .html, .htm, .asp, .txt, and .eml (embedded MIME messages) files. Binary attachments, such as pictures and sounds, are not indexed.

Search results are only as accurate as the last time the index was updated. As the content of public folders or mailbox stores changes, the index must be updated to

reflect the new content. Index updates can be performed manually or automatically on a schedule.

How Outlook Users Access Full-Text Searching

When full-text indexing is deployed, Internet Message Access Protocol version 4rev1 (IMAP4) clients and MAPI clients, such as Microsoft® Outlook®, can conduct full-text searches. For Outlook users, the version of Outlook determines what searching options the user has. In Outlook 2000, on the **Tools** menu, the **Advanced Find** option initiates a full-text search, while the **Find** option initiates a character-based search. In Outlook 2002, both **Find** and **Advanced Find** initiate a full-text search.

Full-Text Search Behaviors

Users accustomed to character-based searches will notice some differences when they perform full-text searches. In particular, recently received e-mail messages do not appear in searches until they have been indexed, and full-text indexing does not search for partial word matches. Other differences between character-based searches and full-text searches are:

- Searching over large amounts of content is much faster with full-text searches than with character-based searches.
- With full-text indexing, searching for words that are commonly used is slower than searching for words that occur less frequently. However, in almost every case, searches are faster and less costly to the server than fixed-string searches.
- With full-text indexing, search results include attachments in addition to messages.
- With full-text indexing, search results include related words, as determined by the word-breaker for the language. For example, word-breaker considers "tester," "tested," and "tests" equivalent, but it considers "testament" equivalent only to "testaments."
- With full-text indexing, pattern-matching does not work; you can search for whole words only. For example, a search for "test" will not include "testament" as a result; a search for "mod" will not include "model" as a result. Searches using wildcard characters (replacing a character with an asterisk) do not work.
- With full-text indexing, "noise" words (word fragments or articles such as "the") are removed from queries. For example, a query for "michael p" searches only for "michael" because "p" is a noise word; a search for "the truth" searches only for "truth."
- With full-text indexing, search results do not include messages received since the last population was performed.
- With full-text indexing, search results are not dynamic. This functionality is displayed in two ways:
 - A view of search results is not immediately updated when a new message arrives that matches the query. This delay is because the new message has not been indexed. It will be indexed on the next incremental crawl or population.

- If the user deletes a message from a folder that is showing in search results, the actual message is deleted, but this view of the folder still shows the deleted message. Repeating the query removes the deleted message from the query results.
- With full-text indexing, when Outlook encounters a comma, it executes an **OR** query. For example, a search for "section, particularly" (without quotation marks) finds all documents with either "section" or "particularly" in the document. To search for the word "section," followed by the word "particularly", put the whole phrase in quotation marks: "section, particularly."

Preparing Your Exchange Environment

Prepare your Exchange environment for full-text indexing by properly configuring your server and ensuring that your Exchange organization is stable.

Preparing Your Server

Before you enable full-text indexing, configure your server for optimal performance by adding sufficient memory and distributing large and frequently accessed files across disks.

Server Configuration

Use a mirrored redundant array of independent disks (RAID) configuration. Microsoft recommends using a RAID-0+1 configuration (also referred to as RAID-1+0 or RAID-10). RAID-5 is not recommended for full-text indexing.

A RAID-0+1 disk array (Figure 1) allows for the highest performance while ensuring redundancy by combining elements of RAID-0 and RAID-1. RAID-1 is a disk array in which two disks are mirrored. RAID-0 is a striped disk array in which each disk is logically partitioned in such a way that a "stripe" runs across all the disks in the array to create a single logical partition. In a RAID-0+1 disk array, data is mirrored to both sets of disks, and then striped across the drives. Each physical disk is duplicated in the array. If you have a six-disk RAID-0+1 disk array, three disks are available for data storage.

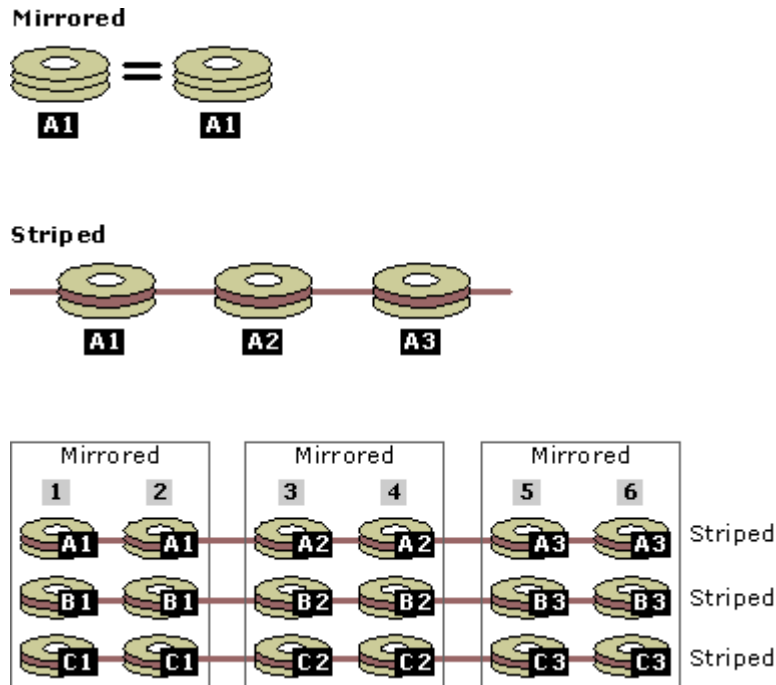


Figure 1 RAID-0+1 disk array

For more information about RAID configurations and Exchange storage solutions, see the "Storage Solutions for Microsoft Exchange 2000 Server" article at <http://www.microsoft.com/Exchange/techinfo/deployment/2000/E2KStorage.asp>

Disk Space Requirements

In Exchange, the Microsoft Search service (MSSearch) powers the full-text indexing. MSearch processes the computer running Exchange Server requests to define and populate indexes for the specified mailbox and public folder stores (Figure 2). MSearch also processes queries initiated by users when they conduct full-text searches.

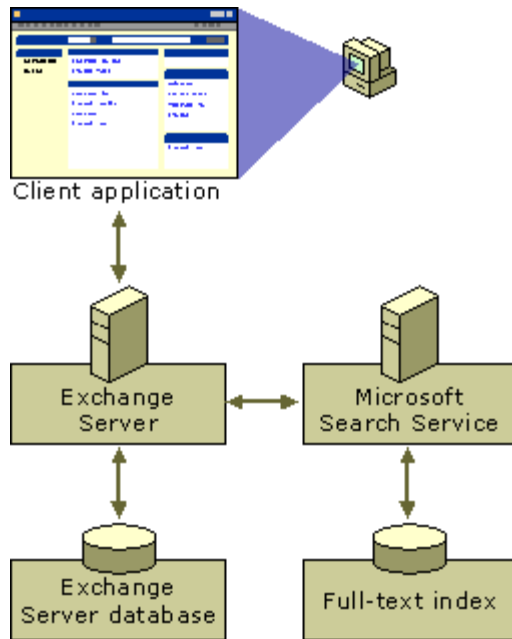


Figure 2 Microsoft Search service (MSearch)

MSearch requires that the disk containing the index, also called a catalog, has 15 percent free disk space at all times. Depending on the types of files you store, the size of your index can range from 10 percent to 30 percent of the size of your database. It is also possible for the index size to exceed the size of the database when users post items to multiple newsgroups. Such items are stored only once in the database, but they are indexed multiple times. In addition to the types of data you plan to index, consider the database growth rate when you determine your disk space requirements.

Memory Requirements

Add an additional 256 megabytes (MB) of RAM to the recommended configuration for a computer running Exchange 2000 Server. Microsoft does not recommend running full-text indexing with less than 512 MB.

File Types and Locations

There are five major categories of full-text indexing files. By default, these files are installed on the system drive, which typically does not have the input/output (I/O) throughput of the RAID array. Arrange the disk locations of these files as shown in Table 1 to optimize the performance of full-text indexing.

- **Catalogs** are the main indexes. There is only one catalog for each mailbox store or public folder store in Exchange.
- **Property store** is a database that contains various properties of items indexed in the catalog. There is only one property store per server.
- **Property store logs** are the log files associated with the property store database.
- **Temporary files** are the files that contain temporary information used by MSearch.

- **Gather logs** are the log files that contain log information for the indexing service. One set of logs exists for each index.

Table 1 Recommended locations for full-text indexing files.

File type	Recommended location	How to specify the location
Catalog	RAID array	Specify a location on the RAID array when you create the catalog using Exchange System Manager. Note If the index was already created elsewhere, use the Catutil tool to move it.
Property store	RAID array	Use the Pstoreuti tool.
Property store logs	RAID array in the same location as the property store	Use the Pstoreuti tool.
Temporary files	RAID array Note On a cluster, place these files on a drive that will not fail over, such as a local drive or a drive on the RAID array or Storage Area Network configured to run only on a designated computer.	Use the SetTmpPath tool.
Gather logs	Leave in the default location, or move to any location you prefer.	Assign the location in the StreamLogsDirectory registry key.

File Placement Tools

You can use the following tools to move specific full-text indexing files to the preferred locations. Run these commands from a command prompt. Detailed procedures for using the tools are located in "Optimizing Full-Text Indexing," later in this article.

- **Pstoreuti** moves the property store and the property store Logs to the drive you specify. The Pstoreuti tool is located in the Program Files\Common Files\System\MSSearch\Bin directory.
- **SetTempPath** moves the temporary directory to the drive you specify. The SetTempPath tool is located in the Program Files\Common Files\System\MSSearch\Bin directory.
- **Catutil** moves the index to the drive you specify. The Catutil tool is located in the Program Files\Common Files\System\MSSearch\Bin directory.

Preparing Your Exchange 2000 Organization

Before you install full-text indexing, verify that your computer running Exchange 2000 Server and your topology is correctly configured and running. If you change your Exchange organization after you install full-text indexing, the index could require a full repopulation. In addition, verify the following:

- Simple Mail Transfer Protocol (SMTP) address configuration is stable and functioning. This configuration affects the URL that is used to index objects.
- The server language is set correctly. To verify the language, open **Control Panel**, double-click **Regional Options**, and then check the language settings for the system. Full-text indexing references the server language specified when

breaking words and stemming—a process that allows a search for “travel” to return “travels,” “traveled,” and “traveling.” Full-text indexing works best when the query language of the client computer matches the language of the files being indexed, and, because the server language is sometimes used for the query language when the client computer language is unknown, it is best for the server language to match the language of most of the documents on the server.

- All servers are properly functioning and connectivity throughout the organization is stable. Perform sufficient tests to ensure that all servers are configured correctly within the organization.

Deploying Full-Text Indexing

After preparing your Exchange environment, use Exchange System Manager to deploy full-text indexing. Deployment involves the following tasks:

- Creating a full-text index
- Optimizing full-text indexing
- Performing a full population
- Setting a schedule for incremental populations
- Enabling full-text indexing queries
- Notifying users

Of these tasks, the most server-intensive is the full population process, which can take from a few minutes for a small database to several days for a large database. However, you can run the population process in the background during business hours without significant impact on system response time for users.

Creating a Full-Text Index

Before you can use full-text indexing, you must create an initial index (catalog).

1. In Exchange System Manager, right-click the mailbox store or public folder store that you want to index.
2. Click **Create Full-Text Index**.
3. A dialog box prompts you to select the location for the index. Specify a place on the RAID array for the index.

Optimizing Full-Text Indexing

Use the following steps to optimize full-text indexing on your computer running Exchange 2000 Server. As stated earlier, by distributing frequently accessed files across a RAID array, you can enhance system performance.

Important This section contains information about editing the registry. Before you edit the registry, make sure you understand how to restore it if a problem occurs. For information about how to restore the registry, view the “Restore the Registry” Help topic in Regedit.exe or Regedt32.exe.

This section references the following tools for moving files:

- **Pstoreutl**, located in Program Files\Common Files\System\MSSearch\Bin
- **SetTempPath**, located in Program Files\Common Files\System\MSSearch\Bin
- **Catutil**, located in Program Files\Common Files\System\MSSearch\Bin

To optimize full-text indexing

1. Move the property store and the property store logs.

When the first index is created on your server, Exchange creates a new property store database on your Exchange system drive. To move the property store database files to your RAID array for improved performance, use the Pstoreutl tool from a command prompt. You need to move the property store and the property store logs only one time for each server because all indexes on a server use the same property store.

To move the property store in a non-cluster environment

- a. From a command prompt, use the Pstoreutl tool to move the database to the new drive.
- b. Restart the MSSearch service.

Example

Your Exchange Property Store database is on drive C and your server name is myserver. You want to move the property store to drive D. From the command line, run the Pstoreutl tool. Use the **-m** command to move the database to the specified location and the **-l** command to change the directory for log files. (Enter the following commands on the same line; they are shown on separate lines below for readability.)

```
pstoreutl.exe ExchangeServer_myserver -m  
d:\exchsrvr\ExchangeServer_myserver\ExchangeServer_myserver.edb -l  
d:\exchsrvr\ExchangeServer_myserver
```

To move the property store in a cluster environment

- a. Leave the MSSearch service running in Control Panel. Use Cluster Administrator to take the MSSearch resource offline.
 - b. Use the Pstoreutl tool to move the database to the new drive. The Exchange data directories are located on the shared disk you specified when you created the Exchange virtual server.
 - c. Use Cluster Administrator to bring the MS Search resource online.
2. Move the temporary directory.

By default, the gather and filter temporary files (also known as temp files) are located on the Exchange system drive, which typically does not have the I/O throughput of the RAID array. Use the SetTempPath tool to move the temporary directory to the RAID array. You need to move this directory only one time for each server because all indexes on a server use the same temporary directory.

To move the MSSearch temporary directory

- From a command prompt, run the SetTempPath tool (see the following example for syntax). (Enter the following command on the same line; it is shown on separate lines below for readability.)

Example

```
cscript "c:\Program Files\Common
Files\System\MSSearch\Bin\settemppath.vbs" d:\temp
```

You can view the current location of the temporary directory at any time by running the SetTempPath script above with no parameters.

Note On a cluster, the full-text index temporary directory must be located on a drive that will not fail over. Make sure you place the temporary directory on a local drive or on a drive on the RAID array or Storage Area Network configured to run only on a designated computer.

3. Move the index (catalog).

The index should be located on the RAID array. If you did not specify this location when you created the index, use the Catutil tool to move it.

To move an index

- a. Stop any active full-text indexing.
- b. Stop and disable the MSSearch service.
- c. From a command prompt, run the Catutil tool.

Note For help using the Catutil tool, go to the command prompt and type `catutil movecat /?`.

Important When you use the Catutil tool, the index moves successfully and functions correctly, but the index location displayed in Exchange System Manager is not updated. This functionality is only a display error that does not affect the normal operation of full-text indexing. You cannot correct the display, but you can check the current location of the index at any time by viewing the following key in the registry:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Search\1.0\Indexer\<application
name>\<index name>\ProjectPath.
```

4. Move the gather logs.

You can choose to leave the gather logs in the default location or you can specify another location using the StreamLogsDirectory registry key. Be sure to specify a valid directory. Full-text indexing does not function if you specify an invalid directory. The Microsoft Search service does not need to be running when you edit the registry key. However, if you edit the registry key while the Microsoft Search service is running, you must restart the service after you make the change for it to take effect.

Warning Using Registry Editor incorrectly can cause serious problems that may require you to reinstall your operating system. Microsoft cannot guarantee that problems resulting from the incorrect use of Registry Editor can be solved. Use Registry Editor at your own risk. For information about how to edit the registry, view the "Change Keys and Values" Help topic in

Registry Editor (Regedit.exe) or the "Add and Delete Information in the Registry" and "Edit Registry Information" Help topics in Regedt32.exe. Note that you should back up the registry before you edit it. If you are running Microsoft® Windows NT® or Microsoft® Windows® 2000, you should also update your Emergency Repair Disk (ERD).

To move the gather logs

- Specify the preferred location for gather logs using the following registry key:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Search\1.0\gather\  
ExchangeServer_<instance>\<index name>\StreamLogsDirectory
```

Note On a cluster, before you change the StreamLogsDirectory registry key, make sure the MSSearch resource is online. Also, make sure you are editing the correct node by verifying which node the group is running on in Cluster Administrator. After you change the registry key, use Cluster Administrator to restart the MSSearch resource by taking it offline, and then bringing it back online.

5. Increase the message size limit.

By default, the index includes messages (including attachments) that are 16 MB or less in size. Therefore, messages with large attachments may be excluded from the index and from the search results of users. To avoid performance problems, Microsoft recommends that you increase this limit to the maximum setting of 4000 MB so that larger messages and attachments are indexed.

The Microsoft Search service does not need to be running when you edit the registry key. However, if you edit the registry key while the Microsoft Search service is running, you must stop and restart the service after you make the change so it takes effect.

To increase the message size limit

- Set the following registry key to 4000 MB:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Search\1.0\  
Gathering Manager\MaxDownloadSize
```

Note On a cluster, before you change the MaxDownloadSize registry key, make sure the MSSearch resource is online. Also, make sure you are editing the correct node by verify the node on which the group is running in Cluster Administrator. After you change the registry key, use Cluster Administrator to restart the MSSearch service by taking it offline, and then bringing it back online.

6. Set up checkpointing (Service Pack 2).

It is strongly recommended that you use the checkpointing script provided with Exchange SP2 to prevent possible indexing problems. If MSSearch terminates abnormally during an incremental crawl (population), some folders and messages may not be properly indexed. Checkpointing remedies this problem by maintaining the following backup files in the catalog directory:

- Two checkpoint record files: <catalog>.chk1.gthr and <catalog>.chk2.gthr.

- Approximately 13 files consisting of the last known complete and uncorrupted set of catalog files stored in a Save subdirectory.

Checkpointing is not turned on by default because it requires a significant amount of additional disk space. The additional file size is approximately 200 bytes for each document in your database. For example, 5,000,000 messages or documents in your database generate checkpointing files totaling 1 gigabyte (GB). The size of these files grows as the number of documents in your database grows. You should ensure that there is sufficient disk space before you run the checkpointing script. It is recommended that at least 15 percent free disk space is available on the disk in which you keep full-text indexing catalogs.

To set up checkpointing

- Ensure that there is sufficient disk space. If necessary, increase the size of the volume or move the catalogs to a larger volume.
- Run the following script from a command prompt:

```
<SystemDrive>:\Program Files\Common
Files\System\MSSearch\Bin\EnableCheckPoints.vbs <APPLICATION>
[CATALOG]
```

Parameter Definitions

- **<APPLICATION>** This is the name of the full-text indexing application. The naming convention for the application is ExchangeServer_<ServerName>. In a stand-alone configuration, <ServerName> is the name of the server. In a clustered environment, <ServerName> is the name of the virtual server.
- **[CATALOG]** This parameter refers to the name of the full-text indexing catalog. You can find the name of the catalog in Exchange System Manager by double-clicking the Exchange store for which a full-text index was created and then double-clicking **Full-Text Indexing**. The property is labeled "Index Name" and the value of the property is the name of the catalog.

Usage

- Specifying zero (0) parameters will output the usage of the script.
- Specifying just the name of the MSSearch application will enable checkpointing for the entire application. This means that all full-text indexes created from this point on will automatically inherit the property. You will have to enable any full-text indexes created prior to this one at a time by specifying the name of the catalog as a parameter to the script.
- Specifying both the name of the MSSearch application and catalog will enable checkpointing for that particular full-text index and no others. This has no affect on future creation of full-text indexes.

Note The only way to disable checkpointing on a catalog, or full-text index, is to delete it and then recreate it.

Examples

- Set up checkpointing on the server Tunis-01 for all new catalogs:

```
D:\Program Files\Common Files\System\MSSearch\Bin\
EnableCheckPoints.vbs ExchangeServer_Tunis-01
```

- Enable checkpointing for an existing catalog on a mailbox store:

D:\Program Files\Common Files\System\MSSearch\Bin\
EnableCheckPoints.vbs ExchangeServer_Tunis-01 privE34F12BB

Performing a Full Population

After you create the index, you must run a full population (also called a crawl) to fill it with data. The resource usage setting for full-text indexing is on the **Full-Text Indexing** tab of the server **Properties**. By default, it is set to **Low**. Microsoft recommends that you use the default setting. A higher setting yields little benefit and could slow down user access to the computer running Exchange Server.

With a low resource usage setting, the population process runs in the background and can be performed during business hours. Threads used during the population use idle processing time. User activities receive priority on the system. Because full-text indexing uses only cycles that would otherwise be idle, it should not significantly slow down user access to the server. Expect CPU usage to approach 100 percent as a normal effect of the population process.

To start a full population

1. Make sure full-text searches are unavailable during full population; otherwise, users will assume that they can conduct full-text searches, but their searches will not return the expected results. To make full-text searches unavailable:
 - a. In Exchange System Manager, right-click the mailbox store or public folder store that you want to index, and then click **Properties**.
 - b. Click **Full-Text Indexing**, and then clear the **This index is currently available for searching by clients** check box.
2. In Exchange System Manager, right-click the mailbox store or public folder store that you want to index.
3. Click **Start Full Population**.

The initial full population can take a long time. With a typical Exchange 2000 configuration, population performance typically ranges from 10 to 20 messages per second. Performance varies based on the hardware configuration, the type and size of messages, and the server resources available. As a result, the total time required for a full population can range from a few minutes for a small database, to several days for a large database. The content language of documents on your server also affects the time the population takes. For example, populating an index on a server containing documents written mostly in East Asian languages can take more than five times longer than for a server containing documents that are written in Western languages. Folders containing Internet newsfeeds can also significantly lengthen population time if the folders contain messages in uuencode format.

Expand the public folder or mailbox store and click **Full-Text Indexing** to view the status of the population. During the initial population, the status is **Crawling**. You can determine that the population has finished by looking at this status or by looking in Event Viewer for MSSearch messages.

Note Do not stop a full population while it is in progress. If you must stop a full population, but intend to rerun it at another time, click **Pause Population** instead of **Stop Population**.

To pause a full population

1. In Exchange System Manager, right-click the mailbox store or public folder store that you want to pause.
2. Click **Pause Population**.

Setting a Schedule for Incremental Populations

Determine how often you want to run an incremental population to update the index. Because an incremental population runs in the background the same way a full population does, frequent updates do not significantly affect system response time for users. Although you should schedule incremental population to occur at least once daily, you may want to schedule more frequent updates because the index is only as current as the last time it was populated. You should also consider the amount of time it takes to complete an incremental population. For example, a typical schedule sets incremental updates at the beginning of each hour. However, if the update lasts more than an hour, the next incremental population begins at the start of the following hour.

Tip Generally, if the mailbox store or public folder store is 6 GB or smaller, you can perform incremental updates hourly. If the store is larger than 6 GB, or the server has high memory usage, you may want to update the index less frequently.

To set the incremental population schedule

1. In Exchange System Manager, go to the mailbox or public folder store that you want to index, right-click it, click **Properties**, and then click the **Full-Text Indexing** tab.
2. In **Update Interval**, select an interval schedule. Typically, you do not need to set the **Rebuild Interval**, which deletes the current index and builds a new one. It is sufficient to set the **Update Interval**.

Enabling Full-Text Indexing Queries

After the initial population and at least one incremental population are complete, enable the use of the index so users can begin conducting full-text searches against the index.

To enable the use of the index

1. In Exchange System Manager, go to the mailbox store or public folder store that you want to enable, right-click it, and then click **Properties**.
2. Click **Full-Text Indexing**, and then click **This index is currently available for searching by clients**.

Notifying and Educating Users

After full-text indexing is installed on a mailbox server, notify users and educate them about what they can expect when they run full-text index searches. See Appendix A later in this paper for a sample e-mail message you can send to your

users to announce the availability of full-text indexing and describe how to use the feature.

Managing Full-Text Indexing

Use the following information to help you manage full-text indexing after deployment. Included are guidelines for determining when to repopulate the index to keep the information current.

Checking the Size of the Index

You can check the size of the index file in the following folder:

```
<driveletter>:\Exchsrvr\ExchangeServer_<servername>\Projects\<indexname>\Build\Indexer\CiFiles
```

Adding Users to an Indexed Server

When you add users to an indexed server, perform an incremental population to add the new mailbox to the index immediately.

Deciding When a New Full Population Is Required

You must fully populate the index in the following cases:

- When a “word-breaker” is changed. (A word-breaker is used by full-text indexing to identify where individual words begin and end in a given text.)
- When noise words are changed.
- When new document format filters are added.
- When the schema file is changed.
- When the SMTP address of the store changes.
- For disaster recovery.

During the population process, the index is still available for full-text queries. The index is unavailable for queries only when you must delete an old index before you re-create it and perform a new full population. This process should be necessary only if the old index is corrupted.

Monitoring Full-Text Indexing

Use System Monitor, Performance Logs and Alerts, and Exchange System Manager to monitor full-text indexing. Use population-related counters to monitor the population process, and use query-related counters to gather statistics on user queries. General performance counters are useful for monitoring system performance and troubleshooting problems related to disk space and paging.

Objects to Monitor in Performance Logs and Alerts

In Performance Logs and Alerts there are five objects with counters that you can monitor to evaluate full-text indexing:

- Microsoft Gatherer
- Microsoft Gatherer Projects
- Microsoft Search
- Microsoft Search Catalogs
- Microsoft Search Indexer Catalogs

Some useful counters from these objects are described in the next section, "Population-Related Counters."

To start Performance Logs and Alerts and create a log to monitor full-text indexing objects

1. On the **Start** menu, point to **Programs**, point to **Administrative Tools**, and then click **Performance**.
2. Expand **Performance Logs and Alerts**, and then click **Counter Logs**.
3. In the details pane, right-click in an empty space, and then click **New Log Settings**.
4. In **Name**, type a name for the log, and then click **OK**.
5. On the **General** tab, click **Add**.
6. In **Performance object**, select one of the five performance objects listed above, and then select one or more counters. Some useful counters are described in the following section.
7. Click **Add**.

Population-Related Counters

Use the following counters with the Microsoft Gatherer and Microsoft Gatherer Projects objects to monitor index population:

- **Microsoft Gatherer: Documents Filtered** This counter is the number of documents that have been filtered and indexed.
- **Microsoft Gatherer: Performance Level** This counter is the system resource usage level. The performance level varies from 1 to 4 as set by Exchange System Manager (1=minimum, 2=low, 3=high, 4=maximum).
- **Microsoft Gatherer: System IO traffic rate** This counter shows the I/O rate used to determine whether to reduce population processing. See the physical disk counters listed in "Disk Usage" later in this paper for more detailed I/O diagnostic information.
- **Microsoft Gatherer: Reason to back off** This counter shows the code describing why the gathering service halted the population.

- 0 - Up and running
- 1 - High IO rate
- 4 - Back off on user activity (by default this is disabled in server install)
- 5 - Battery low (currently, if running on battery, not on AC power)
- 6 - Memory low (less than 5 MB left in paging file)
- **Microsoft Gatherer Projects: Crawl in Progress Flag** This flag contains 0 or 1 and indicates whether a population is running (0=population is running, 1=population is not running).
- **Microsoft Gatherer Projects: Current Crawl is Incremental** This flag contains 0 or 1 and indicates whether the population is incremental or full (0=incremental population, 1=full population).
- **Microsoft Gatherer Projects: Gatherer Paused Flag** This flag contains 0 or 1 and indicates a paused population (0=not paused, 1=paused).
- **Microsoft Gatherer Projects: URLs in History** This flag shows the total number of URLs (folders and documents) known to full-text indexing.
- **Microsoft Gatherer Projects: Waiting Documents** This flag shows the number of documents waiting to be processed by the population. This number increases at the start of a population as new URLs are identified, and then decreases as the population progresses.
- **Microsoft Search Indexer Catalogs: Merge Progress** This counter shows the completion percentage of an index merge.

Query-Related Counters

The following counters for the Microsoft Search Catalogs object contain information about full-text indexing queries. Use them to monitor queries performed against the full-text index.

The Microsoft Search Catalogs object contains the total of the values found in four separate indexes in the Microsoft Search Catalogs object.

- **Microsoft Search Catalogs: Queries** This counter shows the total number of queries performed.
- **Microsoft Search Catalogs: Successful Queries** This counter shows the number of successful queries.
- **Microsoft Search Catalogs: Results** This counter shows the number of rows returned, after trimming, to the scope of the query.
- **Microsoft Search Catalogs: Failed Queries** This counter shows the number of failed queries (for example, those containing noise words).

General Performance-Related Counters

Use the following counters to monitor general performance.

CPU Usage

Use the following counters to monitor CPU usage. Remember that during a full population, CPU usage usually reaches 100 percent.

- **Processor: % Processor Time** This counter ranges from 0 to 100 percent.
- **Process: % Processor Time** This counter ranges from 0 to 100*N processors %. Select from the following instances: "store" is the Exchange Information Store service, "mssearch" is the search process, and "mssdmn" is the indexing process.

Disk Usage

Use the following counters to monitor disk usage. It is essential to have a sufficient amount of free disk space when you run full-text indexing. Serious problems can result if you run out of disk space, such as the index can be corrupted and other system problems can occur. For more information, see "Disk Space Requirements" earlier in this document.

- **Physical Disk: Current Disk Queue Length** This number should not exceed more than 1 or 2 per spindle in the disk system. A higher disk queue length can indicate a bottleneck. This number should frequently return to zero. A disk queue length that rarely or never returns to zero can indicate a bottleneck.
- **Physical Disk: Avg. Disk sec/read** This counter shows the time it takes per disk read. This time is typically about 10 milliseconds. A busy disk has noticeably longer times.
- **Physical Disk: Avg. Disk sec/write** This counter shows the time it takes per disk write, which is typically about 10 milliseconds. However, a RAID array with a write-back cache has a write time of approximately 1 millisecond because the information is held in the controller's cache. Again, as the disk gets busier, this number increases.
- **Physical Disk: Disk Transfers/sec** This counter shows the sum of disk writes and reads per second. Most single spindles have a maximum range of 100 to 150 transfers per second.

Memory Usage

Use the following counters to monitor memory usage:

- **Memory: Available Mbytes** This counter lists free memory on the computer.
- **Process: Virtual Address Space** This counter shows reserved memory (includes virtual allocations).
- **Process: Private Bytes** This counter shows the total allocated memory that is private to this process, for example, the database cache. The counter does not include handles and shared memory.
- **Process: Working Set** This counter shows the allocated memory actually in RAM (equivalent to **Memory Usage** in Task Manager). Pausing a full-text indexing population causes its working set to shrink to approximately 2 MB.

Paging

Use the following counters to monitor paging:

- **Memory: Pages/sec** This counter displays the number of hard pages (to disk) per second. However, more than one page can be processed in a single disk read or write.
- **Memory: Page Writes/sec** This counter displays the number of disk writes per second for paging.
- **Memory: Page Reads/sec** This counter displays the number of disk reads per second for paging.
- **Process: Page Faults/sec** This counter displays hard (to disk) and soft (in memory) page faults per second.

If the **Pages/sec** counter is high (for example, over 100), look at the process generating the most page faults because a correlation is likely. If it is the Exchange Information Store service, check:

- **Database: Database Cache Size** This counter shows the size of the database cache. A small cache can produce high paging because the database cache is swapped back to disk more frequently.

Troubleshooting

This section contains information about how to resolve problems you may encounter with full-text indexing. If you encounter problems with full-text indexing, the Event Viewer and Performance Logs and Alerts are useful troubleshooting tools.

Event Viewer Errors

The Event Viewer is useful for troubleshooting full-text indexing problems; however, there are certain events, described below, that do not necessarily indicate problems.

Event ID 7000: The Indexer Started Successfully

After you use Exchange System Manager to stop an index population, the Indexing Service may incorrectly log several copies of the following event message in the Event Viewer application event log:

```

Event Type: Information
Event Source: Microsoft Search
Event Category: Indexer
Event ID: 7000
Date: date
Time: time
User: N/A
Computer: server_name
Description:
The Indexer started successfully for project
<ExchangeServer_SERVERNAME_priv78F2DC76>

```

Event ID 10006: Catastrophic Failure (Cluster Environment)

When you shut down the indexing service in a cluster environment, you may see the following error:

Event Type: Error
Event Source: Microsoft Search
Event Category: Gatherer
Event ID: 10006
Date: 2/11/2000
Time: 9:44:25 AM
User: N/A
Computer: <servername>
Description:
An error occurred during the online operation for instance <your instance> :
8000ffff - Catastrophic failure

This error is not actually a catastrophic failure. Wait for all services to shut down successfully, and then restart services if necessary. To prevent this error from occurring, use Cluster Administrator to stop clustered resources, not the **Services** application in **Control Panel**. When you stop the service using **Services** in **Control Panel**, the cluster resource manager assumes the resource failed and it either attempts to restart the service or fails over the group.

SMTP and System Administrator Logged as Errors

When the Indexing Service is running, you may receive error messages similar to the following:

```
2b3b1b8 1bed2fc
file:\\.\BackOfficeStorage\server.microsoft.com\MBX\SMTP
(SERVER-{E2E63C70-4129-43F6-9363-6B501433C952}) 8000000c 0
80080005

2cdeb96 1bed2fc
file:\\.\BackOfficeStorage\server.extest.microsoft.com\MBX\System
Attendant 8000000c 0 80080005
```

These error messages can safely be ignored.

Population Process Is Slow

If the population process is slow, Internet newsfeeds may be the cause. Internet newsfeeds may contain uuencoded binaries, which are indexed at a much slower rate than normal messages. When you run a population on a public folder that contains Internet newsfeeds, population time will be significantly lengthened.

Messages with large attachments may also cause performance problems if you have not optimized the maximum download size. The recommended setting is 4000 MB.

Warning Using Registry Editor incorrectly can cause serious problems that may require you to reinstall your operating system. Microsoft cannot guarantee that problems resulting from the incorrect use of Registry Editor can be solved. Use Registry Editor at your own risk. For information about how to edit the registry, view the "Change Keys and Values" Help topic in Registry Editor (Regedit.exe) or the "Add and Delete Information in the Registry" and "Edit Registry Information" Help topics in Regedt32.exe. Note that you should back up the registry before you edit it. If you are running Windows NT or Windows 2000, you should also update your Emergency Repair Disk (ERD).

To change the maximum download size, set the following **DWORD** registry key to 4000 MB:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Search\1.0\Gathering  
Manager\MaxDownloadSize
```

For more information, see "Optimizing Full-Text Indexing" earlier in this document.

Population Process Found in a Paused State

MSSearch pauses a population process if it cannot continue. To verify whether MSearch, rather than an administrator, paused the population, check the event log. MSearch logs an event when it must pause or stop the population. For example, MSearch pauses a population if the disks are too full to add new information to the indexes or the log files. Usually, you can fix the problem (for example, by freeing space on a full drive) and resume the population. New documents added during the pause are not added to the index until the next population.

Note Lack of space on the disk is often the problem, even if it appears that there is plenty of free space. MSearch uses disk space liberally to temporarily unpack large sections of the index to merge new results before recompressing.

Deleted Message Still Visible in Results

You can delete a message from a search result folder. The message is deleted, but the message remains visible in the search result window until you refresh the search.

Wrong Location Is Displayed After Moving Index

If you use the Catutil tool to move the index, the index location displayed in Exchange System Manager will not be updated. The index is successfully moved and functions correctly, but System Manager will incorrectly display the original location of the index. This is only a display error and does not affect the normal operation of full-text indexing. You cannot correct the display, but you can check the current location of the index at any time by viewing the following key in the registry:

```
HKEY_LOCAL_MACHINE \Software\Microsoft\Search\1.0\Indexer\<application  
name>\<i>index name>\ProjectPath.
```

Gather Log Entries

Gather log files are generated during a population. These files contain log information for the indexing service. The files are located in the Program Files\Exchsrvr\ExchangeServer_<servername>\GatherLogs directory. The files have a .gthr extension.

If a particular document fails to be indexed, for any reason, an entry is logged in the gather log file. Each entry lists the file name and error number. To decode this error number, use the Gthrlog tool found in the Program Files\Common

Files\System\MSSearch\Bin directory. At a command prompt, type the following command, where *<filename>* is the name of the .gthr file:

```
cscript gthrlog.vbs <filename>
```

Results from the tool are displayed at the command prompt.

Language Settings Problems

The language settings of individual messages, attachments, the server, and the client computer affect indexing behaviors. This section provides guidelines for these behaviors and scenarios that illustrate the results of mixed language settings.

Full-Text Indexing Guidelines for Mixed Language Settings

The guidelines that govern full-text indexing in mixed-language scenarios are complex. The following guidelines explain how various language settings affect indexing behaviors. Administrators can use these guidelines to help determine the cause of user-reported search problems.

Language Setting of Individual Messages

The language setting of individual messages affects indexing behavior in the following ways:

- If a message is a MAPI message, it has a **Locale ID** property and full-text indexing uses this value to determine which word-breaker to use. This property value comes from the Language setting in Microsoft® Office on the client computer. If full-text indexing cannot find a word-breaker to match the **Locale ID** property, it uses the **Neutral <0>** property.
- If a message is created with Distributed Authoring and Versioning (DAV), it uses the "Accept-Language" header to determine the correct locale.
- If a message has no locale identified (which is often the case with messages from the Internet), the System Locale of the computer running Exchange 2000 Server where full-text indexing is performed is used to determine the word-breaker.

Language Setting of Attachments

The language setting of an attachment affects indexing behavior in the following way:

- If an attachment is a Microsoft Office document, full-text indexing uses the language setting that was used to generate the document.

Language Setting of the Server Running Microsoft Windows 2000

The language setting of the server affects indexing behavior in the following way:

- If the message is non-MAPI (in other words, Internet), its **Locale ID** property is not set and full-text indexing uses the System Locale setting of the server to determine which word-breaker to use.

Language Setting of the Client Computer

The language setting of the client computer affects indexing behavior in the following way:

- When a query is sent from Outlook, the **Locale ID** of the client computer is also sent. If the **Locale ID** of the message does not match the **Locale ID** of the query, the search results are unpredictable.

Note The language of the computer running Exchange Server is irrelevant in this scenario. The client computer setting takes priority.

Full-Text Indexing Behavior in Mixed Language Scenarios

The following scenarios show query behavior of content indexing with various language settings.

All U.S. Language Settings

If you use U.S. language settings in Outlook running on a client computer with U.S. language settings to compose and submit a message to Exchange 2000 running on a Windows 2000 server with U.S. language settings, the following process occurs:

- Full-text indexing indexes the message using the U.S. language setting word-breaker.
- Queries from the client computer with U.S. language settings are processed as expected.

Client computer with Hebrew language settings, U.S. language settings in Office, and Hebrew language settings in Windows 2000

In this example, the client computer is configured as follows:

- Client computer with Hebrew language settings
- Office with U.S. language settings
- Outlook with Hebrew language settings

If you compose a message on the above client computer and submit it to Exchange 2000 with System Locale set to U.S., the following process occurs:

- Full-text indexing uses the U.S. word-breaker to index the message. The **Locale ID** property of the message defaults to U.S. because of the Office settings.
- Queries from the Hebrew client computer fail because the Hebrew document does not have the proper word-breaker applied.

Client computer with Japanese language settings, Japanese language settings in Office, and U.S. language settings in Windows 2000

In this example, the client computer is configured as follows:

- Client computer with Japanese language settings
- Office with Japanese language settings
- Outlook with Japanese language settings

If you compose a message on the above client computer and submit it to Exchange 2000 with System Locale set to U.S., the following process occurs:

- Full-text indexing uses the Japanese word-breaker to index the message.
- Queries from the Japanese client computer succeed because the message was indexed and queried with the same **Locale ID**.

Queries Failing During Initialization

During initialization, in the first few minutes of starting a computer running Exchange Server with full-text indexing, users might get their mail but not get results from queries. This failure to get query results is because MSSearch is loading the index and Exchange is loading the property store. Queries do not return results until these processes are complete.

Missing Performance Counters

Event messages similar to the following indicate that the counters used by System Monitor and Performance Logs and Alerts are missing. If you receive one of these messages, restore the counters by restarting MSSearch.

- Performance monitoring for the Gatherer service cannot be initialized because the counters are not loaded or the shared memory object cannot be opened. This only affects availability of the performance counters. Rebooting the system may fix the problem.
- Performance monitoring cannot be initialized for the Gatherer object because the counters are not loaded or the shared memory object cannot be opened. This only affects availability of the performance counters. Rebooting the system may fix the problem.
- Performance monitoring for the Indexer object cannot be initialized because the counters are not loaded or the shared memory object cannot be opened. Stop and restart the Search service. If this error continues, reinstall the application.

Disk Bottlenecks

To avoid disk bottlenecks, use the following guidelines:

Note For more information, see "Objects to Monitor in Performance Logs and Alerts" earlier in this document.

- Monitor the disk queue length.
 - The queue length is expected to average more than the number of spindles in the RAID array.
 - The length should occasionally drop to zero.
 - The queue should be empty occasionally. Having something always in the queue indicates a problem.
- Average time per disk write and per disk read should be close to expected latency. The system should take roughly 10 milliseconds for a disk read or write.

If configuration has a hardware cache or a RAID controller, the time could be even less.

High Paging

High paging can indicate a memory bottleneck. Check the performance counters listed earlier and monitor them for warning signs. In particular, check the **Memory: Page writes/sec** and **Memory: Page reads/sec** counters.

Additional Resources

- "Storage Solutions for Microsoft Exchange 2000 Server"
<http://www.microsoft.com/Exchange/techinfo/deployment/2000/E2KStorage.asp>

For more information: <http://www.microsoft.com/exchange/>

Did this paper help you? Please give us your feedback. On a scale of 1 (poor) to 5 (excellent), how would you rate this paper?

<mailto:exchdocs@microsoft.com?subject=Feedback: Best Practices for Deploying Full-Text Indexing>

Appendix A. Sample E-Mail Message to Users

You can send the following sample e-mail message to your users to announce the availability of full-text indexing. Note that this message is intended for Microsoft Outlook 2000 users. In Outlook 2000, the Advanced Find option executes a full-text index search, while the Find option executes a character-based search. In Microsoft Outlook 2002, both the Find and the Advanced Find options execute full-text index searches. If your organization uses Outlook 2002, edit the following sample e-mail message accordingly.

Full-text indexing is enabled for all mailboxes on your server. From now on, when you use the **Advanced Find** option in Outlook, you will notice some differences—the biggest of which should be speed!

Full-text indexing searches are faster than the character-based searches you are used to because every word in every e-mail message has been indexed in advance. Instead of searching through the e-mail messages themselves, you are searching through the index.

Please note that Outlook 2000 executes full-text index searches only when you use the **Advanced Find** option on the **Tools** menu. Use the **Find** option to yield a traditional character-based search. It does not use full-text indexing.

The following list describes other differences between full-text indexing and character-based searches in Outlook.

With full-text indexing:

- You get matches for attachments, as well as messages.
- You get matches for related words. For example, full-text indexing considers "tester," "tested," and "tests" equivalent, but "testament" is equivalent only to "testaments."
- You will not get matches that include the newest e-mail messages immediately, because the index is updated on a schedule. Therefore, the newest information is not available until the next update.
- Pattern-matching is not supported. Full-text indexing searches only for whole words. A search for "test" will not include "testament" as a result; a search for "mod" will not include "model." Wild card searching (for example, using an asterisk to represent any string of characters) is not supported. A search for "test*" will not include "test" or other words beginning with "test".
- Noise words (word fragments or articles such as "the") are removed from queries, which means that a query for "michael p" searches only for "michael" because "p" is a noise word; a search for "the truth" searches only for "truth."

Other functions that do not change:

- To search for a combination of words using an **AND** query, simply separate the words with a space. To perform an **OR** query that returns

matches containing either search word, separate the words in the query with a comma.

- To search for an entire phrase, enclose the phrase in quotation marks. For example, a search for "asp files" (with quotation marks) matches against only those two words in succession.
- Without quotation marks, a search for two words returns items that contain both words. A search for "asp files" (without quotation marks) returns items that have both the word "asp" and also the word "file" or variants of those words.



The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2001 Microsoft Corporation. All rights reserved.

Microsoft, Outlook, Windows, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.