Active Data Recovery Software

Active@ UNDELETE

User Guide

Version Number 1.0

Active@ UNDELETE v 1.0 END-USER LICENSE AGREEMENT

~~~~~~

~~~~~~

IMPORTANT-READ CAREFULLY: This End-User License Agreement (“EULA”) is a legal agreement between you (either an individual or a single entity) and The Active Data Recovery Software for the Active@ UNDELETE later referred to as 'SOFTWARE'. By installing, copying, or otherwise using the SOFTWARE you agree to be bound by the terms of this EULA. If you do not agree to the terms of this EULA, do not install or use the SOFTWARE.

WE REQUIRE ALL OUR DEALERS TO PROVIDE EACH PURCHASER WITH FREE DEMO OF THE SOFTWARE TO GET A FULL UNDERSTANDING OF THE CAPABILITIES AND THE EASE OF USE OF THE SOFTWARE. OUR DEALERS HAD TO RECOMMEND YOU TO DOWNLOAD DEMO. WE WON'T ISSUE ANY REFUNDS AFTER PURCHASING FULL VERSION OF THE SOFTWARE.

Active Data Recovery Software may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. The furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

SOFTWARE LICENSE

1. The SOFTWARE is licensed, not sold. Copyright laws and international copyright treaties, as well as other intellectual property laws and treaties protect the SOFTWARE.

2. GRANT OF LICENSE.

(a) FREE DEMO COPY. You may use the full featured DEMO SOFTWARE without charge on an evaluation basis to recover any files having size less than 64Kb. You must pay the license fee and register your copy to recover files bigger than 64Kb in size.

(b) REDISTRIBUTION OF DEMO COPY. If you are using DEMO SOFTWARE on an evaluation basis you may make copies of the DEMO SOFTWARE as you wish; give exact copies of the original DEMO SOFTWARE to anyone; and distribute the DEMO SOFTWARE in its unmodified form via electronic means (Internet, BBS's, Shareware distribution libraries, CD-ROMs, etc.). You may not charge any fee for the copy or use of the evaluation DEMO SOFTWARE itself, but you may charge a distribution fee that is reasonably related to any cost you incur distributing the DEMO SOFTWARE (e.g. packaging). You must not represent in any way that you are selling the software itself. Your distribution of the DEMO SOFTWARE will not entitle you to any compensation from Active Data Recovery Software. You must distribute a copy of this EULA with any copy of the Software and anyone to whom you distribute the SOFTWARE is subject to this EULA.

(c) REGISTERED COPY. After you have purchased the license for SOFTWARE, and have received the registration key and the SOFTWARE distribution package, you are licensed to copy the SOFTWARE only into the memory of the number of computers corresponding to the number of licenses purchased. The primary user of the computer on which each licensed copy of the SOFTWARE is installed may make a second copy for his or her exclusive use on a portable computer. Under no other circumstances may the SOFTWARE be operated at the same time on more than the number of computers for which you have paid a separate license fee. You may not duplicate the SOFTWARE in whole or in part, except that you may make one copy of the SOFTWARE for backup or archival purposes. You may terminate this license at any time by destroying the original and all copies of the SOFTWARE in whatever form. You may permanently transfer all of your rights under this EULA provided you transfer all copies of the SOFTWARE (including copies of all prior versions if the SOFTWARE is an upgrade) and retain none, and the recipient agrees to the terms of this EULA.

3. RESTRICTIONS. You may not reverse engineer, decompile, or disassemble the SOFTWARE, except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation. You may not rent, lease, or lend the SOFTWARE. You may permanently transfer all of your rights under this EULA, provided the recipient agrees to the terms of this EULA. You may not use the SOFTWARE to perform any unauthorized transfer of information (e.g. transfer of files in violation of a copyright) or for any illegal purpose.

4. SUPPORT SERVICES. Active Data Recovery Software may provide you with support services related to the SOFTWARE. Use of Support Services is governed by the Active Data Recovery Software polices and programs described in the online documentation and web site, and/or other Active Data Recovery Software-provided materials, as they may be modified from time to time. Any supplemental software code provided to you as part of the Support Services shall be considered part of the SOFTWARE and subject to the terms and conditions of this EULA. With respect to technical information you provide to Active Data Recovery Software as part of the Support Services, Active Data Recovery Software may use such information for its business purposes, including for product support and development. Active Data Recovery Software will not utilize such technical information in a form that personally identifies you.

5. TERMINATION. Without prejudice to any other rights, Active Data Recovery Software may terminate this EULA if you fail to comply with the terms and conditions of this EULA. In such event, you must destroy all copies of the SOFTWARE.

6. COPYRIGHT. The SOFTWARE is protected by copyright law and international treaty provisions. You acknowledge that no title to the intellectual property in the SOFTWARE is transferred to you. You further acknowledge that title and full ownership rights to the SOFTWARE will remain the exclusive property of Active Data Recovery Software and you will not acquire any rights to the SOFTWARE except as expressly set forth in this license. You agree that any copies of the SOFTWARE will contain the same proprietary notices which appear on and in the SOFTWARE.

7. DISCLAIMER OF WARRANTY. Active Data Recovery Software expressly disclaims any warranty for the SOFTWARE. THE SOFTWARE AND ANY RELATED DOCUMENTATION IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OR MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. THE ENTIRE RISK ARISING OUT OF USE OR PERFORMANCE OF THE SOFTWARE REMAINS WITH YOU.

8. LIMITATION OF LIABILITY. IN NO EVENT SHALL ACTIVE DATA RECOVERY SOFTWARE OR ITS SUPPLIERS BE LIABLE TO YOU FOR ANY CONSEQUENTIAL, SPECIAL, INCIDENTAL, OR INDIRECT DAMAGES OF ANY KIND ARISING OUT OF THE DELIVERY, PERFORMANCE, OR USE OF THE SOFTWARE, EVEN IF ACTIVE DATA RECOVERY SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN ANY EVENT, ACTIVE DATA RECOVERY SOFTWARE'S ENTIRE LIABILITY UNDER ANY PROVOSION OF THIS EULA SHALL BE LIMITED EXCLUSIVELY TO PRODUCT REPLACEMENT.

Active Data Recovery Software reserves all rights not expressly granted here.

Active Data Recovery Software is a registered business name of LSoft Technologies Inc.

Contents

1 OVERVIEW

2 USING ACTIVE@ UNDELETE

3 CONCEPTS

4 UNDERSTANDING ADVANCED UNDELETE PROCESS

5 DESCRIPTION OF UNDELETE FUNCTIONS

Standards Used in This Guide

The following standards are used to provide more concise documentation:

Table 1 User Input

Description	Example	Action
Bold text within square brackets.	Press **[Enter]**. Press **[Y]**	Press the key on the keyboard that corresponds to the message within square brackets.
Bold text and operand within square brackets.	Press **[Ctrl + B]**	Together, press the combination of keys within the square brackets.
Bold text.	Click **OK**	With the mouse pointer, find the icon or button indicated and left-click that icon.

1 OVERVIEW

This chapter gives an overview of **Active@ UNDELETE** application.

Welcome to Active@ UNDELETE

Active@ UNDELETE is a powerful software utility, designed to restore accidentally deleted files and directories. It allows you to recover files that have been deleted from the **Recycle Bin**, as well as those deleted after avoiding the Recycle Bin (e.g. **Shift-Delete**).

Active@ UNDELETE will help you to restore data, residing on hard drives or floppy drives formatted in any of the following file systems:

- FAT12
- FAT16
- FAT32
- NTFS
- NTFS5

It works under all Windows family operating systems:

- Windows 95
- Windows 98
- Windows ME
- Windows NT
- Windows 2000
- Windows XP

Active@ UNDELETE is integrated with **Microsoft Management Console (MMC)** interface, allowing you to use the utility for business applications by incorporating it into a company infrastructure (using **Active Directory**). It has advanced capabilities to work with remote machines. Active@ UNDELETE has a Wizard, that gives expert users full control over the process of data restoration. This advanced recovery feature allows you to restore the contents of partially-overwritten files.

The free evaluation version has full functionality of all features with a limitation only on maximum size of the file being restored.

PROTECT THE DRIVE LOCATION WHERE YOU HAVE ACCIDENTALLY DELETED FILES. Any program that writes data to the disk, even the installation of data recovery software can spoil your sensitive data.

DO NOT SAVE DATA ONTO THE SAME DRIVE THAT YOU FOUND ERASED DATA, WHICH YOU ARE TRYING TO RECOVER! While saving recovered data onto the same drive where sensitive data was located, you can spoil the process of recovering by overwriting table records for this and other deleted entries. It is better to save data onto another logical, removable, network or floppy drive.

IF YOU HAVE AN EXTRA HARD DRIVE, OR OTHER LOGICAL DRIVES THAT ARE BIG ENOUGH, CREATE A DISK IMAGE. A **Disk Image** is a single-file mirror copy of the contents of your logical drive. Backing up the contents of the whole drive - including deleted data - is a good safety precaution in case of failed recovery. Before you start recovering deleted files, create a Disk Image for this drive.

2 USING ACTIVE@ UNDELETE

This chapter describes the features and functions of Active@ UNDELETE.

The Application

Active@ UNDELETE is a powerful software utility, designed to restore files and directories that have been accidentally deleted. This chapter covers the following topics:

- Starting Active@ UNDELETE
- Automatic Data Recovery
- Advanced Data Recovery
- Recovery Tips
- Troubleshooting

What Happened to my Data?

When a file is written to a hard drive, a record of that file is kept in the **Root Table** or **Master File Table (MFT)**. As well, the addresses of **file clusters** are given labels, indicating the clusters are **occupied**. When an existing file is deleted, successful data recovery depends a good deal on the condition of the file clusters. When a file is deleted from a drive location, its clusters are labeled as **unoccupied** and the file entry in **Root Table** or **MFT** indicates the file has been deleted. For more detail on file clusters, see Sectors and Clusters in the following chapter.

After a file has been deleted, the condition of the file clusters depends on whether or not other files have been written to the same drive. There is a chance that the file-writing process may have allocated these clusters and Root Table entries to be overwritten. To recover files successfully, it is strongly recommended to perform the recovery operation immediately after discovering that there are files that have been removed by accident.

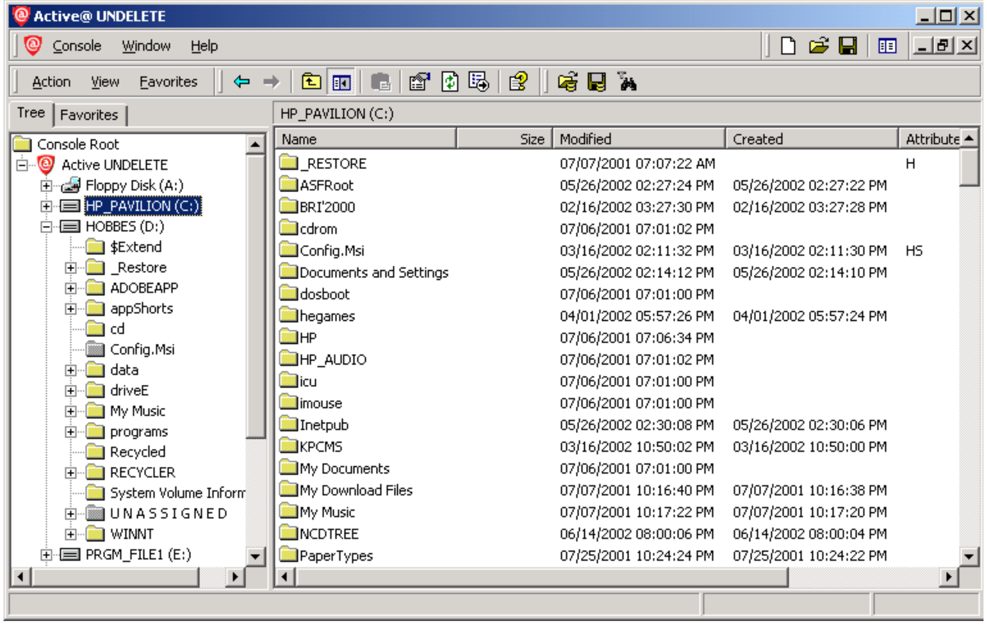
Starting Active@ UNDELETE

After the program has been installed, use Microsoft Start > Programs to open Active@ UNDELETE. When opened, the left pane of the main window holds a list of drives and folders available for scanning. If it is closed, click Active UNDELETE folder to see this list.

Click a drive or folder node. Active@ UNDELETE scans the Root Table or Master File Table and displays information about the disk contents, including recently-deleted files.

The figure below is similar to the main screen:

Figure 2-1 Active@ UNDELETE Screen



Name	Size	Modified	Created	Attribute
_RESTORE		07/07/2001 07:07:22 AM		H
ASFRoot		05/26/2002 02:27:24 PM	05/26/2002 02:27:22 PM	
BRI'2000		02/16/2002 03:27:30 PM	02/16/2002 03:27:28 PM	
cdrom		07/06/2001 07:01:02 PM		
Config.Msi		03/16/2002 02:11:32 PM	03/16/2002 02:11:30 PM	HS
Documents and Settings		05/26/2002 02:14:12 PM	05/26/2002 02:14:10 PM	
dosboot		07/06/2001 07:01:00 PM		
hegames		04/01/2002 05:57:26 PM	04/01/2002 05:57:24 PM	
HP		07/06/2001 07:06:34 PM		
HP_AUDIO		07/06/2001 07:01:02 PM		
icu		07/06/2001 07:01:00 PM		
imouse		07/06/2001 07:01:00 PM		
Inetpub		05/26/2002 02:30:08 PM	05/26/2002 02:30:06 PM	
KPCMS		03/16/2002 10:50:02 PM	03/16/2002 10:50:00 PM	
My Documents		07/06/2001 07:01:00 PM		
My Download Files		07/07/2001 10:16:40 PM	07/07/2001 10:16:38 PM	
My Music		07/07/2001 10:17:22 PM	07/07/2001 10:17:20 PM	
NCDTREE		06/14/2002 08:00:06 PM	06/14/2002 08:00:04 PM	
PaperTypes		07/25/2001 10:24:24 PM	07/25/2001 10:24:22 PM	

If there are many folders and you are not sure where to look for deleted files, Active@ UNDELETE has a filter system that allows you to search for files by name.

- To get more information about disk scanning, see Automatic Data Recovery
- To get more information about file search, see Searching an Unknown Drive Location, in Chapter 5.

The table below describes the icons that appear in the data section of the screen.

Table 2-1 Icons and Descriptions

Icon	Description	Icon	Description
📁	Regular folders	▨	Deleted folder
📄	Regular file	▨	Deleted file

A deleted file or folder that appears as a black icon indicates that deleted file or folder has a poor chance at being recovered. This may be because it has been partially overwritten or completely overwritten.

Automatic Data Recovery

If the deleted file or folder is identified with a gray icon, the entry in the Root Table or MFT and the file clusters have not been overwritten, and you have a good chance to use this method and recover deleted contents successfully. A chain of clusters is defined based on information found in Root Tables or MFT combined with some empirical algorithms. When a folder location is defined for constructing files from this information, the contents of these clusters is written to that folder.

When the folder is recovered, all folder contents - together with subfolders - is restored recursively to the defined folder.

It is simple and intuitive to use this automatic feature for no-nonsense recovery. For example the standard Drag and Drop, or Copy and Paste operations can be used to recover files. Please see Restoring Deleted Files and Folders in Chapter 5 for more details.

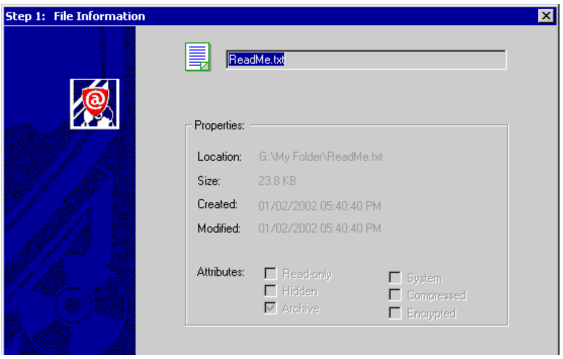
Advanced Data Recovery

Automatic recovery relies completely on the logic implemented in the software and does not provide any flexibility over the process of recovery. If software displays a recovered file with a black icon, the contents of the file may have been damaged or overwritten with other data. The second, overwriting file might also have been deleted. Using advanced features, if the file contents are readable, it is possible to attempt to reconstitute the file manually from the traces of data and then recover the file, partially or completely.

The following steps will take you through a version of Advanced Data Recovery with the Undelete Wizard:

1. Start Active@ UNDELETE.
2. Locate and select a deleted file.
3. To start the Undelete Wizard, click the **Undelete Wizard** icon on the toolbar or right-click the file and click **Undelete Wizard** on the context menu. The **Welcome** screen appears.
4. Read the brief procedure description. If you wish, clear the **Show this dialog next time** checkbox.
5. Click **Next** to continue. The **File Information** screen appears.

Figure 2-2 File Information Screen



Step 1: File Information

ReadMe.txt

Properties:

Location: G:\My Folder\ReadMe.txt

Size: 23.8 KB

Created: 01/02/2002 05:40:40 PM

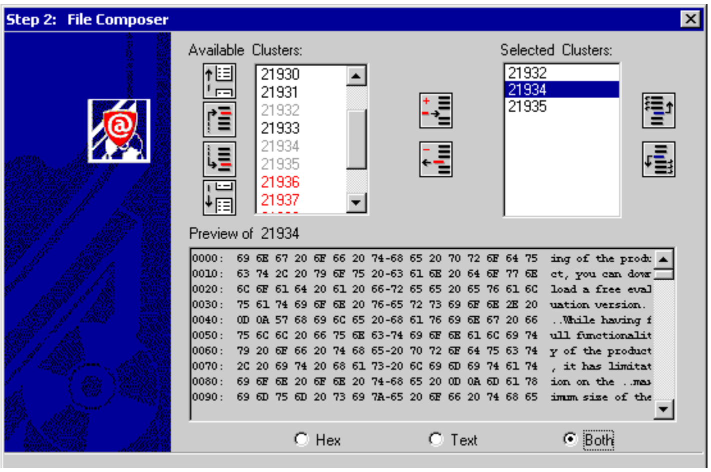
Modified: 01/02/2002 05:40:40 PM

Attributes: □ Read-only □ Hidden ☑ Archive □ System □ Compressed □ Encrypted

6 Some file information is displayed on this screen. Click **Next**.

7 If the file started from a black icon and has poor chances for successful recovery, a warning message will appear. Afterwards the **File Composer** screen appears.

Figure 2-3 File Composer Screen



Step 2: File Composer

Available Clusters:

21930
21931
21932
21933
21934
21935
21936
21937

Selected Clusters:

21932
21934
21935

Preview of 21934

```
0000:  69 6E 67 20 6F 66 20 74-68 65 20 70 72 6F 64 75  ing of the prod
0010:  63 74 2C 20 79 6F 75 20-63 61 6E 20 64 6F 77 6E  ct, you can dow
0020:  6C 6F 61 64 20 61 20 66-72 65 65 20 65 76 61 6C  load a free eval
0030:  75 61 74 69 6F 6E 20 76-65 72 73 69 6F 6E 2E 20  uation version.
0040:  0D 0A 57 68 69 6C 65 20-68 61 76 69 6E 67 20 66  ..While having f
0050:  75 6C 6C 20 66 75 6E 63-74 69 6F 6E 61 6C 69 74  ull functionalit
0060:  79 20 6F 66 20 74 68 65-20 70 72 6F 64 75 63 74  y of the product
0070:  2C 20 69 74 20 68 61 73-20 6C 69 6D 69 74 61 74  , it has limitat
0080:  69 6F 6E 20 6F 6E 20 74-68 65 20 0D 0A 6D 61 78  ion on the ..ma
0090:  69 6D 75 6D 20 73 69 7A-65 20 6F 66 20 74 68 65  imum size of the
```

○ Hex ○ Text ● Both

8 In this screen, you can see and play with the file clusters composing the file, previewing and manipulating them:

Available Clusters Shows the allocated file clusters by number. Click these to move them to the right-hand pane. Clusters occupied by data from other files are colored black. Unoccupied, or free clusters are colored red. Grey-colored clusters are those that have been selected for recovery.

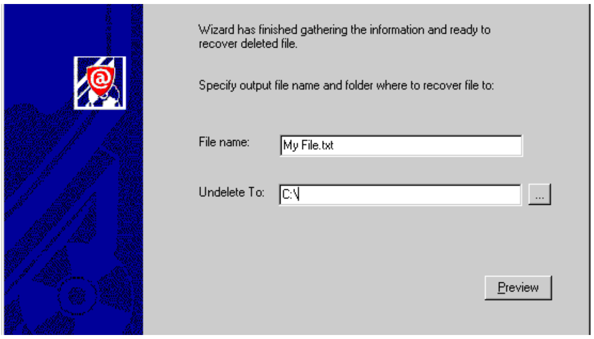
Find **Previous** and **Next** unoccupied cluster using the **Scroll Up** or **Scroll Down** icons. Contents of clusters selected in this box will be displayed in the **Preview** pane, below.

Selected Clusters Displays clusters as they are selected from the left-hand pane. Assemble these clusters in order so that the contents of the file makes logical sense.

Select clusters by number and click **Add** or **Remove** icons to edit the contents of this pane. Similarly, click **Move Up** or **Move Down** icons to re-arrange the order of these clusters.

Once assembly of the file body is complete, click **Next** to complete the recovery process. The **Finish** screen appears.

Figure 2-4



Wizard has finished gathering the information and ready to recover deleted file.

Specify output file name and folder where to recover file to:

File name: My File.txt

Undelete To: C:\

9 On this screen, change the path and name of the file to be recovered. Save a recovered file to a location different from that of the original data. For more information on this topic, please read the next section.

If Automatic Data Recovery fails to recover file contents properly, changing the **File name** to a simple format like *.txt, *.log or *.rtf can make the file easier to recover.

10 **Preview** the file using default viewer. After all output parameters have been defined, click **Finish** to complete recovery process.

For more information on sectors and clusters, please see Hard Disk Drive Basics in Chapter 3.

Choose Restore Location Carefully

It is important to restore files to a logical drive, removable drive, floppy disk or network drive other than the location from which the deleted file data is being recovered. For safety of the recovered data, the default settings of the utility do not allow writing the restored file into the same drive as the deleted file. The reason for this is that there is a chance that a newly-created file might overwrite data that is being recovered, or destroy the contents of other deleted files.

If there is no other choice, it is possible to configure the utility to write the recovered file to the same location as the original data.

NOTE: If saving a recovered file to an original location, save it with a different file name. If file contents are not recovered properly, this step can help in differentiating clusters of data in the Advanced Data Recovery method.

Follow these steps to allow writing to the same drive:

1 In the main screen, click **Action**.

2 Click **Properties**. The **Active Undelete Properties** screen appears.

3 Click the **Preferences** tab.

4 In the **Recovery** section of this screen, enable the checkbox **Allow to the same drive**.

Recovery Tip

Please read through these tips for best recovery practice.

IF POSSIBLE, DO NOT WRITE ANYTHING ONTO THE DRIVE CONTAINING YOUR IMPORTANT DATA. To install Active@ UNDELETE, you must write files onto a hard drive. If the erased data is really important to you, and your computer operates with only one drive, follow these steps:

1 Go to a second computer and install Active@ UNDELETE.

2 At the first computer, turn the power off and remove the power cable from the power source.

3 Ground yourself to avoid static electric charge.

4 Open your computer case and remove the entire hard drive with deleted files from the bracket.

5 Take the removed hard drive to the second computer and attach it so that the second computer has access to its own drive and to the drive with deleted files.

Troubleshooting

The following scenarios are provided from help service offered to actual customers.

Situation A

When starting Active@ UNDELETE, I get an error message that reads "**Access is Denied**". The main screen does not display any drives. I'm running Windows NT 4.0.

Possible Cause Windows NT 4.0, Windows 2000 and Windows XP have a built-in security feature. You might be logged into the system with an account that does not belong to the **Administrators** group. To recover erased data, low-level access to hardware is required. With administrative privileges, you will receive permissions that allow installing new software and running it.

Solution If Active@ UNDELETE was installed, uninstall it. Log off as the current user and log on with a profile that is member of the **Administrators** group. Install software again and run it.

Situation B

I have Windows NT 4.0. I have installed MMC but when I try to run it, I see an error that says:

```
"ActiveUndelete.MSC" is not a Microsoft Management
Console Document or cannot be run.
```

Possible Cause #1: Your **Microsoft Management Console** has version 1.1 or less. You can check the version of MMC by clicking **Help -> About** from the main menu command bar.

Possible Cause #2: Windows NT needs to be rebooted after MMC installation to activate file association for all documents having the extension ".MSC" with newly installed Microsoft Management Console components.

Solution for #1: Uninstall software and explicitely mark "Microsoft Management Console" component when you install it again. Proper version of MMC should be downloaded and installed.

Solution for #2: Reboot the machine and try running software again.

3 CONCEPTS

This chapter describes some basic concepts that might help when unerasing data.

Hard Disk Drive Basics

A hard disk is a sealed unit containing a number of **platters** in a stack. Hard disks may be mounted in a horizontal or a vertical position. In this description, the hard drive is mounted horizontally.

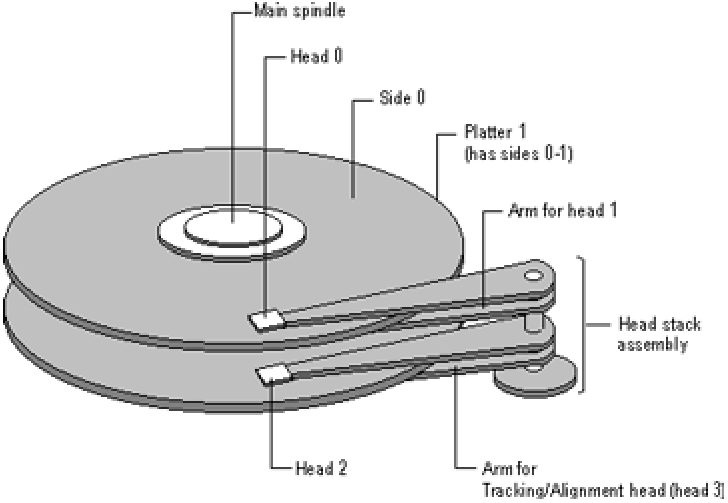
Electromagnetic read/write **heads** are positioned above and below each platter. As the platters spin, the drive heads move in toward the center surface and out toward the edge. In this way, the drive heads can reach the entire surface of each platter.

Making Tracks

On a hard disk, data is stored in thin, concentric bands. A drive head, while in one position can read or write a circular ring, or band called a **track**. There can be more than a thousand tracks on a 3.5-inch hard disk. Sections within each track are called **sectors**. A sector is the smallest physical storage unit on a disk, and is almost always 512 bytes (0.5 kB) in size.

The figure below shows a hard disk with two platters.

Figure 3-1 Parts of a Hard Drive



The structure of older hard drives (i.e. prior to Windows 95) will refer to a **cylinder/ head/ sector** notation. A cylinder is formed while all drive heads are in the same position on the disk. The tracks, stacked on top of each other form a cylinder. This scheme is slowly being eliminated with modern hard drives. All new disks use a translation factor to make their actual hardware layout appear continuous, as this is the way that operating systems from Windows 95 onward like to work.

To the operating system of a computer, tracks are logical rather than physical in structure, and are established when the disk is low-level formatted. Tracks are numbered, starting at 0 (the outermost edge of the disk), and going up to the highest numbered track, typically 1023, (close to the center). Similarly, there are 1,024 cylinders (numbered from 0 to 1023) on a hard disk.

The stack of platters rotate at a constant speed. The drive head, while positioned close to the center of the disk reads from a surface that is passing by more slowly than the surface at the outer edges of the disk. To compensate for this physical difference, tracks near the outside of the disk are less-densely populated with data than the tracks near the center of the disk. The result of the different data density is that the same amount of data can be read over the same period of time, from any drive head position.

The disk space is filled with data according to a standard plan. One side of one platter contains space reserved for hardware track-positioning information and is not available to the operating system. Thus, a disk assembly containing two platters has three sides available for data. Track-positioning data is written to the disk during assembly at the factory. The system **disk controller** reads this data to place the drive heads in the correct sector position.

Sectors and Clusters

A sector, being the smallest physical storage unit on the disk, is almost always 512 bytes in size because 512 is a power of 2 (2 to the power of 9). The number 2 is used because there are two states in the most basic of computer languages - on and off.

Each disk sector is labelled using the factory track-positioning data. Sector identification data is written to the area immediately before the contents of the sector and identifies the starting address of the sector.

The optimal method of storing a file on a disk is in a **contiguous** series, i.e. all data in a stream stored end-to-end in a single line. As many files are larger than 512 bytes, it is up to the file system to allocate sectors to store the file's data. For example, if the file size is 800 bytes, two 512 k sectors are allocated for the file. A **cluster** is typically the same size as a sector. These two sectors with 800 bytes of data are called two clusters. They are called clusters because the space is reserved for the data contents. This process protects the stored data from being over-written. Later, if data is appended to the file and its size grows to 1600 bytes, another two clusters are allocated, storing the entire file within four clusters.

If contiguous clusters are not available (clusters that are adjacent to each other on the disk), the second two clusters may be written elsewhere on the same disk or within the same cylinder or on a different cylinder - wherever the file system finds two sectors available. A file stored in this non-contiguous manner is considered to be **fragmented**. Fragmentation can slow down system performance if the file system must direct the drive heads to several different addresses to find all the data in the file you want to read. The extra time for the heads to travel to a number of addresses causes a delay before the entire file is retrieved.

Cluster size can be changed to optimize file storage. A larger cluster size reduces the potential for fragmentation, but increases the likelihood that clusters will have unused space. Using clusters larger than one sector reduces fragmentation, and reduces the amount of disk space needed to store the information about the used and unused areas on the disk.

Understanding Microsoft Management Console (MMC)

Microsoft Management Console (MMC) hosts administrative tools that you can use to maintain networks, computers, services, and other system components.

The MMC administrative tools (called **MMC Consoles**) manage the hardware, software, and network components in the Windows system. MMC is a feature of the **Windows 2000** operating system, and it can be used on Windows NT, Windows 95, and Windows 98 operating systems. In addition, MMC functions are used in conjunction with several software applications designed to run in the Windows environment.

MMC does not perform administrative functions, but hosts tools that do. The primary type of tool you can add to a console is called a **snap-in**. Other items that you can add include:

- ActiveX controls
- Links to Web pages and folders
- Links to the Windows Task Manager, views, and tasks

In general, there are two ways to use MMC:

1. **Author Mode -** Creating consoles by modifying existing MMC Consoles or writing new ones
2. **User Mode -** Administration with existing MMC Consoles

For more information about the differences between user and author mode, see MMC Console Access Options.

The MMC Window

If your current operating system has MMC Console functions available, they are located on the **Start Button - Programs** menu or in the **Administrative Tools** folder in **Control Panel**. To open MMC, click **Start**, click **Run**, and then type **mmc** and press **[Enter]**.

An MMC window appears divided into two panes. The left pane contains two tabs labeled **Tree** and **Favorites**. The Tree tab, also called the **console tree**, displays the items that are available in a given console. The right pane is called the **details pane**. It shows details about - and command functions pertaining to - items selected in the console tree. The details pane can display many types of information including Web pages, graphics, charts, tables, and columns.

At the top of each MMC window is a command menu bar for opening or creating new MMC Consoles. Click the **New Console** icon or click **Console - New** from the command menu.

When a new MMC window is opened, it displays command toolbar and drop-down menus, similar to, but separate from those of the main MMC window. In addition, there is a status bar at the bottom of the window and a description bar along the top of the details pane.

In the console window, you can assemble and configure a new console and then work with the tools in the console. After items are added to a console, you can hide any of the tool bars to prevent users from making unnecessary changes to the console.

Snap-Ins

A snap-in is the basic component of an MMC console. Snap-ins always work from within a console and they do not run by themselves. When you install a component that has a snap-in associated with it on a computer running Windows, the snap-in is available to anyone creating a console on that computer (unless restricted by a user policy).

Stand-Alone and Extension Snap-Ins

MMC supports two types of snap-ins:

- Stand-alone snap-ins (snap-ins)
- Extension snap-ins (extensions)

Add a snap-in to a console tree first, without adding another item. An extension is always added to a snap-in or to another extension. When extensions are enabled for a snap-in, they operate on the objects controlled by the snap-in, such as a computer, printer, modem, or other device. When a snap-in or extension is added to a console, it may appear as a new item in the console tree, or it may add context menu items, toolbars, property pages, or wizards.

Adding Snap-Ins

Add snap-ins one-at-a-time, or many-at-a-time along with other items. Multiple instances of a single snap-in may be added to the same console to adjust different computers or to repair a damaged console.

Each time a new instance of snap-in is added to a console, any variables for the snap-in are set at default values until configured otherwise. For example, when a snap-in is configured to manage a remote computer, the unique configuration is not transferred when adding a second instance of the snap-in. The second instance will be set to default values.

As a general rule, snap-ins can be added only to the local computer you are using. In Windows 2000, however, a computer that is part of a domain, can download any snap-ins that are not locally installed, but that are available in the **Active Directory** directory service.

For more information about distributing software by using Active Directory in Windows 2000, see **Windows 2000 Server Help**.

Taskpad Views and Tasks

Taskpad views are pages to which you can add views of the details pane of a console, as well as shortcuts to functions both inside and outside a given console. A taskpad view might make it easier for novice users to perform their jobs.

Use these shortcuts to run tasks such as starting wizards, opening property pages, performing menu commands, running command lines, and opening Web pages. Configure a taskpad view so that it contains all the tasks a given user might need to perform a specific function. In addition, multiple taskpad views can be created in a console, so that tasks can be grouped by function or by user.

Here is an example:

The user is required to create a document in Microsoft Word (doc), then create and review an Adobe Acrobat (pdf) copy of the document. Add appropriate shortcuts to a taskpad view and then hide the console tree. The user can begin using the applications before they are familiar with the structure of the system directory tree. You may also use taskpad views to make complex tasks easier. For instance, if a user must frequently perform a task that involves multiple snap-ins and other tools, you can present tasks in a single location that open or run the necessary dialog boxes, property pages, command lines, and scripts.

MMC Console Access Options

When building a **custom console**, assign the console one of two general access options: **author mode** or **user mode**. There are, in turn, three levels of user mode, so that there are four options for default access to a console:

- Author mode
- User mode - full access
- User mode - limited access, multiple window
- User mode - limited access, single window

Author mode access is not necessary for users who do not create or change MMC consoles. A system administrator can configure user profile settings to prevent users from opening MMC in author mode.

Configure these options in the **Options** dialog box in MMC. Assign **author mode** to a console to grant full access to all MMC functionality, including the ability to add or remove snap-ins, create new windows, create taskpad views and tasks, add items to the Favorites list, and view all portions of the console tree.

By assigning one of the user mode options, authoring functions are restricted. For example, the **User mode - full access** option allows access to the console tree and all management commands, and restricts access to adding or removing snap-ins or changing the console properties.

Changes made in **author mode** are saved differently from those made in **user mode**. When closing the console in author mode, there is a prompt to save changes. When closing a console in user mode, changes will be not be saved.

MMC Resources

For more information about MMC, consult the following resources:

- The Microsoft Management Console integrated Help (click **Help Topics** from the **Help** menu)
- The Microsoft Management Console Web site: (http://www.microsoft.com/) contains information for snap-in authors and developers
- The Windows 2000 Web site (http://www.microsoft.com/) provides a tutorial for creating and customizing MMC consoles
- Download the latest version of MMC (http://www.microsoft.com/)

Understanding Distributed COM (DCOM)

The Distributed Component Object Model (DCOM) is a protocol that enables software components to communicate directly over a network in a reliable, secure, and efficient manner. Previously called “Network OLE,” DCOM is designed for use across multiple network transports, including Internet protocols such as HTTP.

The Component Object Model (COM) can make distributed applications secure without any security-specific coding or design in either the client or the component. Just as the COM programming model hides a component's location, it also hides the security requirements of a component. The same binary code that works in a single-machine environment, in which security may be of no concern, can be used securely in a distributed environment.

Running DCOMCNFG

Configure DCOM using a number of different utilities including MS DOS **registry editor**, **OLE View** utility, or **DCOMCNFG** configuration utility.

DCOMCNFG is a utility included with Microsoft Windows NT operating system and is used to configure various security-specific settings in the registry. To engage the program, follow these steps:

1. Click **Start** in Microsoft Windows.
2. Click **Run** from the program menu. The Run screen appears.
3. Type **dcomcnfg** in the **Open** field and click **OK**. The **Distributed COM Configuration Properties** screen appears.

If DCOMCNFG does not produce the properties screen, download and install it from the **Microsoft** web site. Configure an application's COM properties before attempting to communicate over the network. The DCOMCONFG utility can be used to set permissions for all or selected applications in the following categories:

- Distributed components. By default, distributed components are enabled.
- Application location or path.
- Server applications.
- User accounts. The client application uses this account to start processes and gain access to resources on the server computer.
- Connections between applications, for example, packet encryption. Both computers that are running the client and the server applications must be configured for a distributed environment with the DCOMCONFG utility, as follows:
  - **Client Application**. Specify the path to the server application. When a COM client application is used, it makes a request to a server application, which could be running on a different computer.
  - **Server Application**. Specify the user accounts that will have permission to use or start and run the server application.

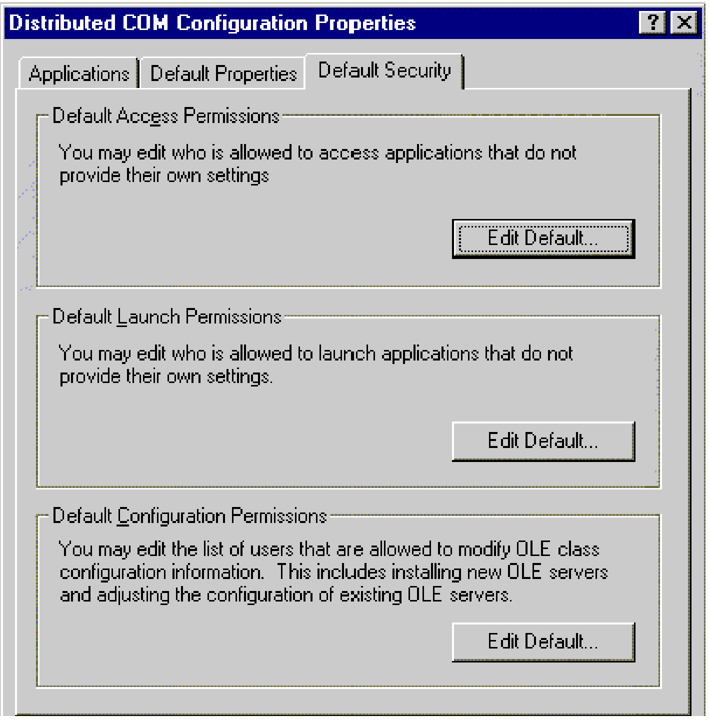
Using DCOMCNFG

Changes made here are recorded in the system registry under **HKEY_LOCAL_MACHINE\Software\Microsoft\OLE**. Follow these steps to configure DCOM:

1 When DCOMCNFG starts, it displays the **Distributed COM Configuration Properties** dialog box. This dialog box has three tabs: **Default Security**, **Default Properties**, and **Applications**.

2 Click the **Default Security** tab. This screen displays three sections: **Access**, **Launch**, and **Configuration Permissions**.

Figure 3-2 Default Security Screen


Distributed COM Configuration Properties
?
Applications
Default Properties
Default Security
Default Access Permissions
Edit Default...
Default Launch Permissions
Edit Default...
Default Configuration Permissions
Edit Default...

You may edit who is allowed to access applications that do not provide their own settings

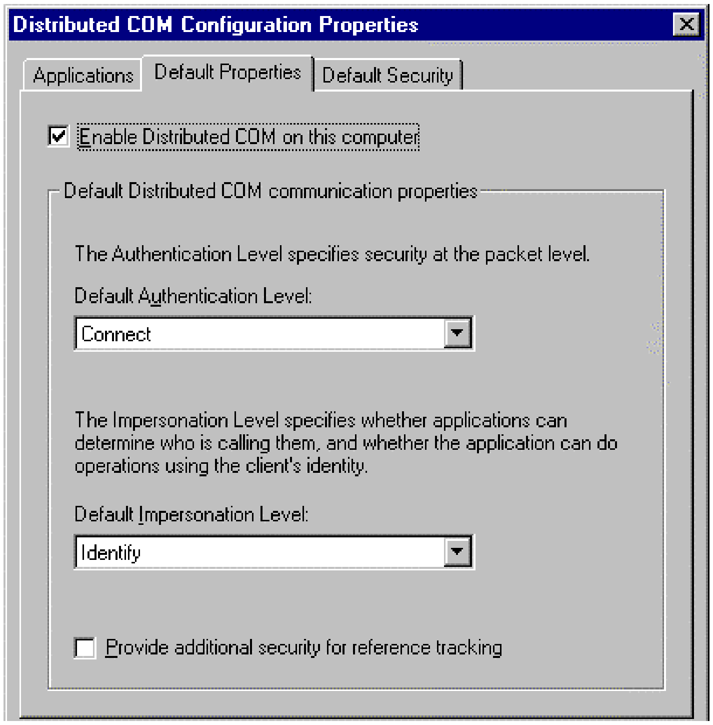
You may edit who is allowed to launch applications that do not provide their own settings.

You may edit the list of users that are allowed to modify OLE class configuration information. This includes installing new OLE servers and adjusting the configuration of existing OLE servers.

3 Click the corresponding **Edit Default** button to make changes.

4 Click the **Default Properties** tab. The **Default Properties** screen appears.

Figure 3-3 Default Properties Screen



☑ Enable Distributed COM on this computer

Default Distributed COM communication properties

The Authentication Level specifies security at the packet level.

Default Authentication Level: Connect

The Impersonation Level specifies whether applications can determine who is calling them, and whether the application can do operations using the client's identity.

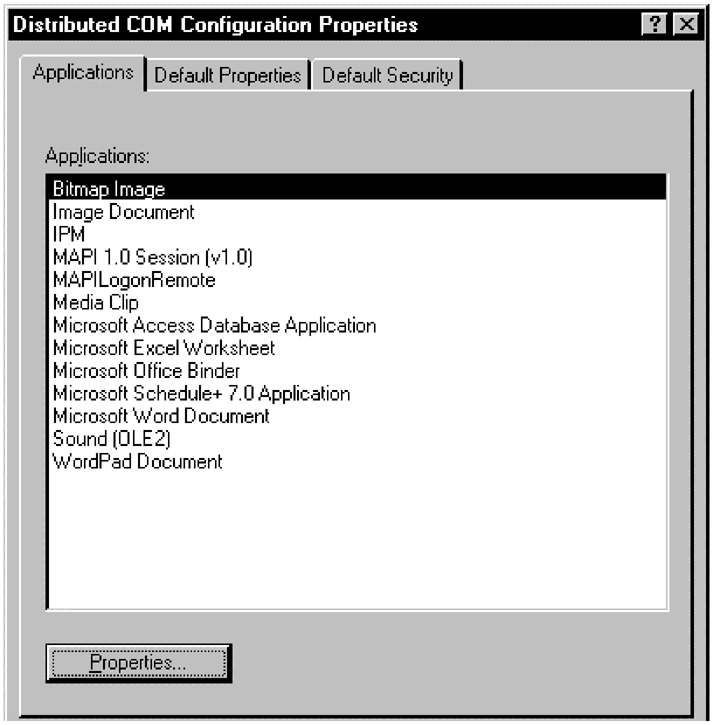
Default Impersonation Level: Identify

□ Provide additional security for reference tracking

5 Check **Enable Distributed COM on This Computer** if you want clients on other machines to access COM objects running on this machine. Selecting this option sets the **HKEY_LOCAL_MACHINE\Software\Microsoft\OLE\EnableDCOM** value to **Y**.

6 Click the **Applications** tab. The **Applications** screen appears.

Figure 3-4

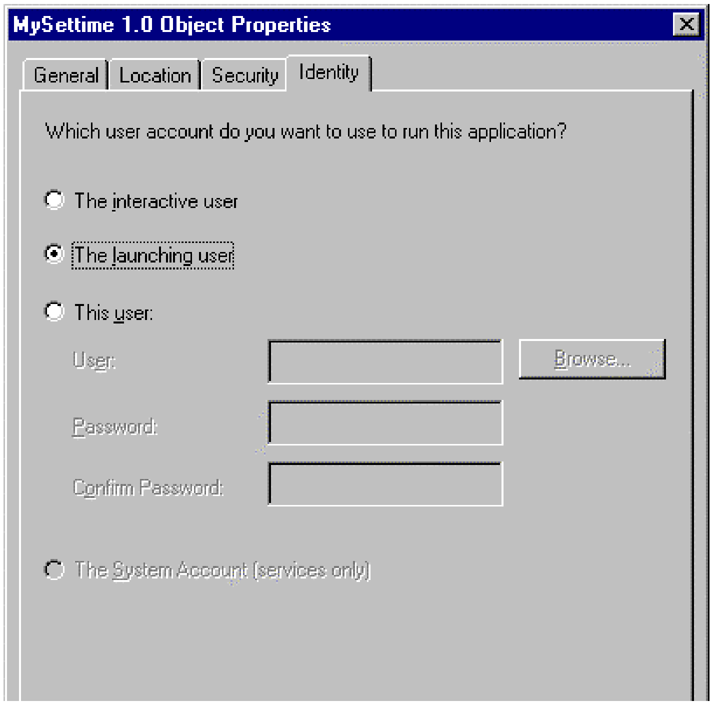


Applications:

Bitmap Image
Image Document
IPM
MAPI 1.0 Session (v1.0)
MAPILogonRemote
Media Clip
Microsoft Access Database Application
Microsoft Excel Worksheet
Microsoft Office Binder
Microsoft Schedule+ 7.0 Application
Microsoft Word Document
Sound (OLE2)
WordPad Document

7 To change the settings for a particular object, select the application from the list and click the **Properties** button. This action displays the **Object Properties** dialog box for the selected application.

Figure 3-5



Which user account do you want to use to run this application?

○ The interactive user

● The launching user

○ This user:

User:

Password:

Confirm Password:

○ The System Account (services only)

8 The **Object Properties** dialog box has four tabs. The table below provides a description:

Table 3-1 Object Properties Tabs

Tab Name	Description
General	Confirms the name of the application.
Location	Specifies where the application should run when a client creates the instance of server application. If **Run Application on the Following Computer** is enabled, and a computer name is entered, a **RemoteServerName** value is added under the APPID for that application.
Security	Similar to the **Default Security** tab found in the **Distributed COM Configuration Properties** dialog box, except that these settings apply only to the current application. Again, the settings are stored under the APPID for that object.
Identity	Identifies which user is used to run the application.

Setting Machine-Wide Security

All applications do not provide access security. Set machine-wide security when you want to apply common security settings for all users. **Dcomcnfg.exe** makes it easy to set default values in the registry that apply to all applications on a machine.

It is important to understand that if a client or server utility explicitly sets a security level that affects the system process-wide, default settings in the registry will be ignored. Also, if Dcomcnfg.exe is used to specify security settings for a specific process, the default machine settings will be overridden by the settings for the adjusted process.

When enabling machine-wide security, set the authentication level to a value other than **None**. Set **launch** and **access** permissions. Setting the default **impersonation level**, and **reference tracking** settings are optional

The following topics in this section provide step-by-step procedures:

- Default Authentication Level
- Launch Permissions
- Access Permissions
- Impersonation Level
- Reference Tracking
- Enabling and Disabling DCOM

Default Authentication Level

The authentication level tells COM the level at which the client is authenticated. Various levels of protection are offered, ranging from **no protection** to **full encryption**.

Choose a setting using **Dcomcnfg.exe**, by completing the following steps.

1. Run Dcomcnfg.exe.
2. Click the **Default Properties** tab.
3. From the **Default Authentication Level** list box, choose any value other than **None**.
4. To continue setting machine properties, click the **Apply** button. The new authentication levels will be applied.
5. When complete, click **OK** to apply the changes and exit Dcomcnfg.exe.

up

Launch Permissions

The launch permissions set with **Dcomcnfg.exe** explicitly grants or denies permission to launch any server that does not provide its own launch-permission settings.

Add or remove users or groups, specifying permission.

Follow these steps to set launch permissions for a machine:

1. In **Dcomcnfg.exe**, click the **Default Security** tab.
2. On the **Property** page, click **Edit Default** button in the **Default Launch Permissions** area.
3. To remove users or groups, select the user or group you want to remove and click the **Remove** button. When you have finished removing users and groups, click **OK**.

4 To add a user or group, click the **Add** button.

Enter a user name in the **Add Names** text box or select it from the user database **Names** list box and click the **Add** button.

5 Select access type from the **Type of Access** list box, (**Allow Launch** or **Deny Launch**).

6 Add all users with the same type of access by entering names or choosing them from the list until finished, then click **OK** to apply changes.

7 To add users with different user access, repeat steps 5 and 6.

up

Access Permissions

Set access permissions for access to servers that do not provide their own access permissions. Add or remove users or groups, specifying permission.

*Note: When setting access permissions, ensure that **SYSTEM** is included in the list of users. Granting access permissions to **Everyone**, includes SYSTEM implicitly.*

The process of setting access permissions for a machine is similar to setting launch permissions, as described above. Here is a summary of the steps:

1 On the **Default Security** property page, click **Edit Default**.

2 To remove users or groups, select them and click **Remove**. When complete, click **OK**.

3 To add a user or group, click the **Add** button.

Enter a user name in the **Add Names** text box or select it from the user database **Names** list box and click the **Add** button.

4 Select access type from the **Type of Access** list box, (**Allow Launch** or **Deny Launch**).

5 Add all users with the same type of access by entering names or choosing them from the list until finished, then click **OK** to apply changes.

6 To add users with different user access, repeat steps 5 and 6.

up

Impersonation Level

The impersonation level, set by the client, determines the amount of authority given to the server to act on the client's behalf. For example, when the client has set its impersonation level to **delegate**, the server can access local and remote resources as the client, and the server can cloak over multiple machine boundaries (provided the cloaking capability is set).

To set the impersonation level for a machine:

1 In **Dcomcnfg**, click the **Default Properties** tab.

2 From the **Default Impersonation Level** list box, click impersonation level you want.

3 To continue setting machine properties, click the **Apply** button. The new authentication levels will be applied.

4 When complete, click **OK** to apply the changes and exit Dcomcnfg.exe.

up

Reference Tracking

This function asks COM to do additional security checks and to keep track of information that will keep objects from being released too early. Keep in mind that these additional checks are expensive.

Use the following steps to enable or disable reference tracking. To set reference tracking for a machine:

1 In **Dcomcnfg**, click the **Default Properties** tab.

2 **Enable** or **disable** the **Provide additional security for reference tracking** check box near the bottom of the page.

3 To continue setting machine properties, click the **Apply** button. The new authentication levels will be applied.

4 When complete, click **OK** to apply the changes and exit Dcomcnfg.exe.

up

Enabling and Disabling DCOM

When a computer is part of a network, the DCOM wire protocol enables COM objects on that computer to communicate with COM objects on other computers. You can disable DCOM for a particular computer, but doing so will disable all communication between objects on that computer and objects on other computers.

Disabling DCOM on a computer has no effect on local COM objects. COM still looks for launch permissions that you have specified. If no launch permissions have been specified, default launch permissions are used. Even if you disable DCOM, if a user has physical access to the computer, they could launch a server on the computer unless you set launch permissions not to allow it.

Warning If you disable DCOM on a remote computer, you will not be able to remotely access that computer afterwards to re-enable DCOM. To re-enable DCOM, you will need physical access to that computer.

To manually enable (or disable) DCOM for a computer:

1 In **Dcomcnfg**, click the **Default Properties** tab.

2 **Enable** or **disable** the **Enable Distributed COM on this Computer** check box.

3 To continue setting machine properties, click the **Apply** button. The new authentication levels will be applied.

4 When complete, click **OK** to apply the changes and exit Dcomcnfg.exe.

up

Process-wide Security

To enable security for a specific application if that application has security needs that are different from those required by other applications on the machine. For instance, you might decide to use machine-wide settings for your applications that require a low level of security and a higher level of security for a specific application.

Security settings in the registry that apply to a specific application are sometimes not used. For example, the application-wide settings set in the registry using **Dcomcnfg.exe** will be overridden if a client sets security explicitly for an interface proxy. When enabling security for an application, several settings may need to be modified. These include the following

- Authentication Level
- Location
- Launch Permissions
- Access Permissions
- Identity
- Browsing the User Database

Authentication Level

To enable security for an application, you must set an authentication level other than None. The authentication level tells COM how much authentication protection is required, and it can range from authenticating the client at the first method call to encrypting parameter states fully. To set an application's authentication level:

1. On the Applications property page in Dcomcnfg.exe, select the application and click the Properties button (or double-click the selected application).
2. On the General page, select an authentication level other than (None) from the Authentication Level list box.
3. To continue setting machine properties, click the **Apply** button. The new authentication levels will be applied.
4. When complete, click **OK** to apply the changes and exit Dcomcnfg.exe.

Location

Determine the computer location on which the application will run. Choose to run it on the machine where the data is located, on the machine used to set the location, or on another specified machine, using these steps:

1. In **Dcomcnfg**, click the **Application** tab.
2. Click the application. Click the **Properties** button (or double-click the selected application).
3. On the **Location** page, select one or more check boxes that correspond to locations where you want the application to run.

   If more than one check box is enabled, COM uses the first one that applies. If **Dcomcnfg** is being run on the server machine, always select **Run Application On This Computer**.
4. To continue setting machine properties, click the **Apply** button. The new authentication levels will be applied.
5. When complete, click **OK** to apply the changes and exit Dcomcnfg.exe.

up

Launch Permissions

In **Dcomcnfg** set permissions to launch a particular server. Add or remove users or groups, specifying permission. Follow these steps:

1. In **Dcomcnfg**, click the **Applications** tab and click the application.
2. Click **Properties** (or double-click the selected application).
3. On the **Security** property page, click **Use custom launch permissions**. Click **Edit**.
4. To remove users or groups, select them and click **Remove**. When finished, click **OK**.
5. To add a user or group, click the **Add** button.

   Enter a user name in the **Add Names** text box or select it from the user database **Names** list box and click the **Add** button.
6. Select access type from the **Type of Access** list box, (**Allow Launch** or **Deny Launch**).
7. Add all users with the same type of access by entering names or choosing them from the list until finished, then click **OK** to apply changes.
8. To add users with different user access, repeat steps 5 and 6.

up

Access Permissions

Set access permissions to grant or deny access to the methods of a particular server. Add or remove users or groups, specifying permission.

*Note: When setting access permissions, ensure that **SYSTEM** is included in the list of users. Granting access permissions to **Everyone**, includes SYSTEM implicitly.*

The process of setting access permissions for an application is similar to setting launch permissions. The steps are as follows:

1. In **Dcomcnfg**, click the **Applications** tab and click the application.
2. Click **Properties** (or double-click the selected application).
3. On the **Security** property page, click **Use custom access permissions**. Click **Edit**.
4. To remove users or groups, select them and click **Remove**. When finished, click **OK**.
5. To add a user or group, click the **Add** button.

   Enter a user name in the **Add Names** text box or select it from the user database **Names** list box and click the **Add** button.
6. Select access type from the **Type of Access** list box, (**Allow Access** or **Deny Access**).
7. Add all users with the same type of access by entering names or choosing them from the list until finished, then click **OK** to apply changes.
8. To add users with different user access, repeat steps 5 and 6.

up

Identity

An application's identity is the account that is used to run the application. The identity can be that of the user that is currently logged on (the interactive user), the user account of the client process that launched the server, a specified user, or a service.

Use **Dcomcnfg** following these steps:

1. In **Dcomcnfg**, click the **Applications** tab and click the application.
2. Click **Properties** (or double-click the selected application).
3. On the **Identity** property page, select the appropriate option button. If you choose **This User:**, you must type in the user name and password.
4. To continue setting machine properties, click the **Apply** button. The new authentication levels will be applied.
5. When complete, click **OK** to apply the changes and exit Dcomcnfg.exe.

Browsing the User Database

Browse the user database in **Dcomcnfg** to find the fully qualified user name for a particular user (e.g. to identify a user to change permissions). To browse the user database:

1. In the **List Names From** list box, select the domain containing the user or group you want to add.
2. To see the users that belong to the selected domain, click **Show Users**.
3. To see the members of a particular group, select the group and click **Show Members**.
4. If you cannot locate the user or group you want to add, click **Search**. The **Find Account** dialog box appears.

   a Click a domain for searching (or click **Search All**).

   b Type a user name.

   c Click **Search** to execute the search.

up

Windows 95 and Windows 98 Security Setup

In a network configuration, the default behavior of COM is to make a **secure call** and then default to an **un-secure call**. In a Windows 95- or Windows 98-only network, change COM's default behavior so that only un-secure calls can be made.

If there is a Windows NT or Windows 2000 domain, both Windows 95 and Windows 98 can provide authentication and authorization using a pass-through security mechanism and no changes are necessary.

To change this behavior, the following tasks must be completed:

- Set the authentication level for call security to be NONE for both client and server.
- On activation, the client must specify an authentication level of NONE.
- Disable reference tracking.

Windows 95/98 COM Servers

When a Windows 95/98 COM server is used to serve objects to remote clients, make sure to:

1 Run **Regedit**. Verify that the **EnableDCOM** and **EnableRemoteConnections** registry keys under HKLM\Software\Microsoft\OLE are set to **Y** on the server machine.

EnableDCOM must be set to **Y** to enable any distributed COM functionality.

EnableRemoteConnections must be set to **Y** to let the machine act as a server.

2 On a Windows 95 computer, manually start the server.

Windows 95 does not support launching servers through COM.

Windows 95/98 client and Windows NT Server

The authentication level is negotiated as follows:

If you have a Windows 95/98 client with authentication level **Connect** and a Windows NT server object with authentication level **Encrypt**, COM will try to use Encrypt for calls in both directions.

Since Windows 95/98 cannot receive calls at Encrypt, the Windows NT computer cannot call the Windows 95/98 machine. Thus both the client and server have to set the authentication level to the lowest common value allowable for any call in any direction. Similarly, if you have two processes, one with a logon token and the other with an impersonation token, and you set the authentication level to **none** in the second, it still won't be able to call the first if its authentication level is not **none**.

For a detailed explanation of these issues, see the MSDN Knowledge Base article Q174024 entitled FAQ: DCOM95 Frequently Asked Questions.

For a detailed explanation of security issues, see the MSDN Knowledge Base Security in COM.

DCOM Resources

For more information about DCOM, try the following resources:

- The Microsoft COM Technologies Web site at the Microsoft web site (http://www.microsoft.com/) contains information for understanding DCOM principles.
- The Microsoft Developers Network Web site at the Microsoft web site (http://msdn.microsoft.com/) provides more information about DCOM architecture and configuration, useful for developers and system administrators.
- Download the latest version of DCOM from Microsoft web site (http://www.microsoft.com/).

4 Understanding Advanced Undelete Process

This chapter describes various processes of the application.

Overview

The process to undelete a file consists of scanning a drive or folder to discover deleted entries, as listed in the Root Folder (File Allocation Table) or Master File Table (NT File System). Once a deleted entry has been found, a chain of file clusters is defined for recovery and then the contents of these clusters is written to the newly created file.

Different file systems maintain their own specific logical data structures, however basically each file system follows these rules:

- A list or catalog of file entries and deleted files is kept. This list can be scanned for entries marked as deleted.
- For each catalog entry, a list of data cluster addresses is kept. From the deleted file entry, a set of clusters composing the file can be located.

After finding the deleted file entry and assembling the associated set of clusters, the data from them can be read and copied to another location.

It is important to note, however that not every deleted file can be recovered. To be successful, it is important to try every method available. In order to try every method, sometimes it is necessary to push ahead, even though going on assumed information, such as:

- In order to begin, assume that the file entry still exists (that is has not been overwritten with other data). The sooner a recovery or undelete attempt is made, the better. This reduces the chance that new files have written on top of the deleted data, and improves the chance that the file can be recovered.
- The second assumption is that the file entry in the Table is reliable enough to point to the location of the file clusters. In some cases (specifically in Windows XP, and on larger FAT32 volumes) the operating system damages the Table file entries immediately after a file is deleted. The important first data cluster becomes invalid and further restoration might not be possible.
- The third assumption is that the file data clusters are intact (they have not been overwritten with other data). The fewer write operations that have been performed on the drive where deleted file used to reside, the more chances that the space occupied by data clusters of the deleted file have not been used for other data storage.

In general, here’s what to do immediately after data loss:

1 PROTECT THE DRIVE LOCATION WHERE YOU HAVE ACCIDENTALLY DELETED FILES. Any program that writes data to the disk, even the installation of data recovery software can spoil your sensitive data.

2 DO NOT SAVE DATA ONTO THE SAME DRIVE THAT YOU FOUND ERASED DATA, WHICH YOU ARE TRYING TO RECOVER! While saving recovered data onto the same drive where sensitive data was located, you can spoil the process of recovering by overwriting table records for this and other deleted entries. It is better to save data onto another logical, removable, network or floppy drive.

The rest of this chapter contains step-by-step examples on these topics:

Disk Scanning

Defining the Chain of Clusters

Recovering the Chain of Clusters

Disk Scanning

Disk Scanning is the process of low-level assessment of all entries in the **Root Folders** on FAT12, FAT16, FAT32 or in **Master File Table** (MFT) on NTFS, NTFS5.

The objective is to find and display deleted entries. In spite of different file and folder entry structure in the different file systems, both of them have common file attributes, as listed in the table below:

Table 4-1 Common File Attributes

FAT12, FAT16, FAT32	NTFS, NTFS5
Root File Allocation Table	Master File Table
Table Location	Table Location
File Size	File Size
Table Structure	Table Structure
File Name	File Name
Date/Time Created	Date/Time Created
Attributes	Attributes
Existing/Deleted Status	Existing/Deleted Status

Given that a any file table, folder or file has a location, size and predefined structure, it is possible to scan data on the drive from the beginning to the end, reading the actual data, not only the record kept in the file table. That information can be displayed and assessed.

NOTE Deleted entries are marked differently depending on the file system. For example, in FAT any deleted entry, file or folder is marked with the ASCII symbol 229 (OxE5) as the first symbol of the entry file name. On NTFS a deleted entry has a special attribute in the file header that points to whether the file has been deleted or not.

Scanning a FAT16 Folder

In this example, the folder contains 3 entries, one of which is deleted.

1 The first entry is an existing folder called **MyFolder**. (long entry and short entry)

```
0003EE20 41 4D 00 79 00 46 00 6F 00 6C 00 0F 00 09 64 00 AM.y.F.o.l....d.
0003EE30 65 00 72 00 00 00 FF FF FF FF 00 00 FF FF FF FF e.r...yyyy..yyyy
0003EE40 4D 59 46 4F 4C 44 45 52 20 20 20 10 00 4A C4 93 MYFOLDER ..JA″
0003EE50 56 2B 56 2B 00 00 C5 93 56 2B 02 00 00 00 00 00 V+V+..A″V+......
```

2 The second entry is a deleted file called **MyFile.txt** (long entry and short entry)

```
0003EE60 E5 4D 00 79 00 46 00 69 00 6C 00 0F 00 BA 65 00 aM.y.F.i.l...?e.
0003EE70 2E 00 74 00 78 00 74 00 00 00 00 00 FF FF FF FF ..t.x.t.....yyyy
0003EE80 E5 59 46 49 4C 45 20 20 54 58 54 20 00 C3 D6 93 aYFILE TXT .AO″
0003EE90 56 2B 56 2B 00 00 EE 93 56 2B 03 00 33 B7 01 00 V+V+..i″V+..3..
```

3 The third one is an existing file called **Setuplog.txt**. (only short entry)

```
0003EEA0 53 45 54 55 50 4C 4F 47 54 58 54 20 18 8C F7 93 SETUPLOGTXT .??″
0003EEB0 56 2B 56 2B 00 00 03 14 47 2B 07 00 8D 33 03 00 V+V+....G+..?3..
0003EEC0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
0003EED0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
```

The first symbol of the deleted file entry (MyFile.txt) is marked with **E5** symbol, so **Disk Scanner** can assume that this entry has been deleted.

Scanning an NTFS5 Folder (Windows 2000):

For our drive we have input parameters:

- Total Sectors 610406
- Cluster size 512 bytes
- One Sector per Cluster
- MFT starts from offset 0x4000, non-fragmented
- MFT record size 1024 bytes
- MFT Size 1968 records

From this information, we can read through all 1968 MFT records, starting from the absolute offset 0x4000 on the volume, looking for the deleted entries. We are most interested in MFT entry 57 having offset 0x4000 + 57 * 1024 = 74752 = 0x12400 because it contains our recently deleted file **"My Presentation.ppt"**

Below MFT record number 57 is displayed:

```
Offset    0  1  2  3  4  5  6  7     8  9  A  B  C  D  E  F
00012400 46 49 4C 45 2A 00 03 00    9C 74 21 03 00 00 00 00   FILE*...?t!.....
00012410 47 00 02 00 30 00 00 00    D8 01 00 00 00 04 00 00   G...0...O.......
00012420 00 00 00 00 00 00 00 00    05 00 03 00 00 00 00 00   ................
00012430 10 00 00 00 60 00 00 00    00 00 00 00 00 00 00 00   ....`...........
00012440 48 00 00 00 18 00 00 00    20 53 DD A3 18 F1 C1 01   H....... SY?.nA.
00012450 00 30 2B D8 48 E9 C0 01    C0 BF 20 A0 18 F1 C1 01   .0+OHeA.A? .nA.
00012460 20 53 DD A3 18 F1 C1 01    20 00 00 00 00 00 00 00   SY?.nA. .......
00012470 00 00 00 00 00 00 00 00    00 00 00 00 02 01 00 00   ................
00012480 00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00   ................
00012490 30 00 00 00 78 00 00 00    00 00 00 00 00 00 03 00   0...x...........
000124A0 5A 00 00 00 18 00 01 00    05 00 00 00 00 00 05 00   Z...............
000124B0 20 53 DD A3 18 F1 C1 01    20 53 DD A3 18 F1 C1 01   SY?.nA. SY?.nA.
000124C0 20 53 DD A3 18 F1 C1 01    20 53 DD A3 18 F1 C1 01   SY?.nA. SY?.nA.
000124D0 00 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00   ................
000124E0 20 00 00 00 00 00 00 00    0C 02 4D 00 59 00 50 00   .........M.Y.P.
000124F0 52 00 45 00 53 00 7E 00    31 00 2E 00 50 00 50 00   R.E.S.~.1...P.P.
```

```
00012500 54 00 69 00 6F 00 6E 00   30 00 00 00 80 00 00 00   T.i.o.n.0...^...
00012510 00 00 00 00 00 00 02 00   68 00 00 00 18 00 01 00   ........h.......
00012520 05 00 00 00 00 00 05 00   20 53 DD A3 18 F1 C1 01   ........ SY?.nA.
00012530 20 53 DD A3 18 F1 C1 01   20 53 DD A3 18 F1 C1 01   SY?.nA. SY?.nA.
00012540 20 53 DD A3 18 F1 C1 01   00 00 00 00 00 00 00 00   SY?.nA........
00012550 00 00 00 00 00 00 00 00   20 00 00 00 00 00 00 00   ........ .......
00012560 13 01 4D 00 79 00 20 00   50 00 72 00 65 00 73 00   ..M.y. .P.r.e.s.
00012570 65 00 6E 00 74 00 61 00   74 00 69 00 6F 00 6E 00   e.n.t.a.t.i.o.n.
00012580 2E 00 70 00 70 00 74 00   80 00 00 00 48 00 00 00   ..p.p.t.^...H...
00012590 01 00 00 00 00 00 04 00   00 00 00 00 00 00 00 00   ................
000125A0 6D 00 00 00 00 00 00 00   40 00 00 00 00 00 00 00   m.......@.......
000125B0 00 DC 00 00 00 00 00 00   00 DC 00 00 00 00 00 00   .U.......U......
000125C0 00 DC 00 00 00 00 00 00   31 6E EB C4 04 00 00 00   .U......1neA....
000125D0 FF FF FF FF 82 79 47 11   00 00 00 00 00 00 00 00   yyyy,yG.........
000125E0 00 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ................
000125F0 00 00 00 00 00 00 00 00   00 00 00 00 00 00 03 00   ................
...............
00012600 00 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ................
```

MFT Record has pre-defined structure. It has a set of attributes defining any file of folder parameters. MFT Record begins with standard File Record Header (first bold section, offset 0x00):

- “FILE” identifier (4 bytes)
- Offset to update sequence (2 bytes)
- Size of update sequence (2 bytes)
- LogFile Sequence Number (LSN) (8 bytes)
- Sequence Number (2 bytes)
- Reference Count (2 bytes)
- Offset to Update Sequence Array (2 bytes)
- Flags (2 bytes)
- Real size of the FILE record (4 bytes)
- Allocated size of the FILE record (4 bytes)
- File reference to the base FILE record (8 bytes)
- Next Attribute Id (2 bytes)

The most important information in this block is the file state, either **deleted** or **in-use**. If Flags field (in red color) has **bit 1** set, it means that file is **in-use**. In our example it is zero, which means the file is **deleted**.

Starting from 0x48, we have **Standard Information Attribute** (second bold section):

- File Creation Time (8 bytes)
- File Last Modification Time (8 bytes)
- File Last Modification Time for File Record (8 bytes)
- File Access Time for File Record (8 bytes)
- DOS File Permissions (4 bytes) 0x20 in our case Archive Attribute

Following standard attribute header, we have **File Name Attribute** belonging to DOS name space, short file names, (third bold section, offset 0xA8) and again following standard attribute header, we have **File Name Attribute** belonging to Win32 name space, long file names, (third bold section, offset 0x120):

- File Reference to the Parent Directory (8 bytes)

- File Modification Times (32 bytes)
- Allocated Size of the File (8 bytes)
- Real Size of the File (8 bytes)
- Flags (8 bytes)
- Length of File Name (1 byte)
- File Name Space (1 byte)
- File Name (Length of File Name * 2 bytes)

In our case from this section we can extract file name, “My Presentation.ppt”, **File Creation** and **Modification** times, and **Parent Directory Record number**. Starting from offset 0x188, there is a non-resident **Data** attribute (green section).

- Attribute Type (4 bytes) (e.g. 0x80)
- Length including header (4 bytes)
- Non-resident flag (1 byte)
- Name length (1 byte)
- Offset to the Name (2 bytes)
- Flags (2 bytes)
- Attribute Id (2 bytes)
- Starting VCN (8 bytes)
- Last VCN (8 bytes)
- Offset to the Data Runs (2 bytes)
- Compression Unit Size (2 bytes)
- Padding (4 bytes)
- Allocated size of the attribute (8 bytes)
- Real size of the attribute (8 bytes)
- Initialized data size of the stream (8 bytes)
- Data Runs ...

In this section we are interested in **Compression Unit size** (zero in our case means non-compressed), **Allocated** and **Real size** of attribute that is equal to our file size (0xDC00 = 56320 bytes), and **Data Runs** (see the next topic).

Defining the Chain of Clusters

To reconstruct a file from a set of clusters, we need to define a chain of clusters. Here are the steps:

1. Scan the drive to locate and identify data.
2. One-by-one, go through each file cluster (NTFS) or each free cluster (FAT) that we presume belongs to the file
3. Continue chaining the clusters until the size of the cumulative total of clusters approximately equals the total size of the deleted file. If the file is fragmented, the chain of clusters will be composed of several extents (NTFS), or select probable contiguous clusters and bypass occupied clusters that appear to have random data (FAT).

The location of these clusters can vary depending on file system. For example, a file deleted in a FAT volume has its first cluster in the Root entry; the other clusters can be found in the File Allocation Table. In NTFS each file has a **_DATA_** attribute that describes "data runs". Disassembling data runs reveals **extents**. For each extent there is a **start cluster offset** and a **number of clusters in extent**. By enumerating the extents, the file's cluster chain can be assembled.

The clusters chain can be assembled manually, using low-level disk editors, however it is much simpler using a data recovery utility, like **Active@ UNERASER**.

Defining a Cluster Chain in FAT16

In the previous topic, we were examining a sample set of data with a deleted file named **MyFile.txt**. This example will continue with the same theme.

The folder we scanned before contains a record for this file:

```
0003EE60 E5 4D 00 79 00 46 00 69   00 6C 00 0F 00 BA 65 00   aM.y.F.i.l...?e.
0003EE70 2E 00 74 00 78 00 74 00   00 00 00 00 FF FF FF FF   ..t.x.t.....yyyy
0003EE80 E5 59 46 49 4C 45 20 20   54 58 54 20 00 C3 D6 93   aYFILE TXT .AO"
0003EE90 56 2B 56 2B 00 00 EE 93   56 2B 03 00 33 B7 01 00   V+V+..i"V+..3..
```

We can calculate size of the deleted file based on root entry structure. Last four bytes are 33 B7 01 00 and converting them to decimal value (changing bytes order), we get 112435 bytes. Previous 2 bytes (03 00) are the number of the first cluster of the deleted file. Repeating for them the conversion operation, we get number 03 - this is the start cluster of the file.

What we can see in the File Allocation Table at this moment?

```
Offset    0  1  2  3  4  5  6  7     8  9  A  B  C  D  E  F
00000200 F8 FF FF FF FF FF 00 00   00 00 00 00 00 00 08 00   oyyyyy..........
00000210 09 00 0A 00 0B 00 0C 00   0D 00 FF FF 00 00 00 00   ..........yy....
00000220 00 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ................
```

Zeros! And it is good in our case - it means that these clusters are free, i.e. most likely our file was not overwritten by another file's data. Now we have chain of clusters 3, 4, 5, 6 and we are ready to recover it.

Some explanations:

- We started looking from offset 6 because each cluster entry in FAT16 takes 2 bytes, our file starts from 3rd cluster, i.e. 3*2=6.
- We considered 4 clusters because cluster size on our drive is 32 Kb, our file size is 112, 435 bytes, i.e. 3clusters*32Kb = 96Kb plus a little bit more.
- We assumed that this file was not fragmented, i.e. all clusters were located consecutively. We need 4 clusters, we found 4 free consecutive clusters, so this assumption sounds reasonable, although in real life it may be not true.

Note: In many cases data cannot be successfully recovered, because the cluster chain cannot be defined. This will occur when another file or folder is written on the same drive as the one where the deleted file is located. Warning messages about this fact will be displayed while recovering data using Active@ UNDELETE.

Defining a Cluster Chain in NTFS

When recovering in NTFS, a part of DATA attributes called **Data Runs** provides the location of file clusters. In most cases, DATA attributes are stored in the Master File

Table (MFT) record. Finding the MFT record for a deleted file will most likely lead to the location of the cluster's chain.

In example below the DATA attribute is marked with a green color. Data Runs inside the DATA attribute are marked as Bold.

```
Offset    0  1  2  3  4  5  6  7     8  9  A  B  C  D  E  F
00012580 2E 00 70 00 70 00 74 00    80 00 00 00 48 00 00 00   ..p.p.t._...H...
00012590 01 00 00 00 00 00 04 00    00 00 00 00 00 00 00 00   ................
000125A0 6D 00 00 00 00 00 00 00    40 00 00 00 00 00 00 00   m.......@.......
000125B0 00 DC 00 00 00 00 00 00    00 DC 00 00 00 00 00 00   .U.......U......
000125C0 00 DC 00 00 00 00 00 00    31 6E EB C4 04 00 00 00   .U......1neA....
000125D0 FF FF FF FF 82 79 47 11    00 00 00 00 00 00 00 00   yyyy,yG.........
```

Decrypting Data Runs

Decrypting data runs can be accomplished using the following steps:

1 First byte (0x31) shows how many bytes are allocated for the length of the run (0x1 in the example case) and for the first cluster offset (0x3 in our case).

2 Take one byte (0x6E) that points to the length of the run.

3 Pick up 3 bytes pointing to the start cluster offset (0xEBC404).

4 Changing bytes order we get first cluster of the file 312555 (equals 0x04C4EB).

5 Starting from this cluster we need to pick up 110 clusters (equals 0x6E).

6 Next byte (0x00) tells us that no more data runs exist.

7 Our file is not fragmented, so we have the only one data run.

8 Lastly, check to see if there is enough information (size of the file). Cluster size is 512 bytes. There are 110 clusters, 110*512 = 56,320 bytes. Our file size was defined as 56,320 bytes, so we have enough information now to recover the file clusters.

Recovering the Chain of Clusters

After the cluster chain is defined, the final task is to read and save the contents of the defined clusters to another place, verifying their contents. With a chain of clusters and standard formulae, it is possible to calculate each **cluster offset** from the beginning of the drive. Formulae for calculating cluster offset vary, depending on file system. Starting from the calculated offset, copy a volume of data equal to the size of the chain of clusters into a newly-created file.

To calculate the cluster offset in a FAT drive, we need to know:

- Boot sector size
- Number of FAT-supported copies
- Size of one copy of FAT
- Size of main root folder
- Number of sectors per cluster
- Number of bytes per sector

NTFS format defines a linear space and calculating the cluster offset is simply a matter of multiplying the cluster number by the cluster size.

Recovering Cluster Chain in FAT16

This section continues the examination of the deleted file **MyFile.txt** from previous topics. By now we have chain of clusters numbered 3, 4, 5 and 6 identified for recovering. Our cluster consists of 64 sectors, sector size is 512 bytes, so cluster size is: 64*512 = 32,768 bytes = 32 Kb.

The first data sector is 535 (we have 1 boot sector, plus 2 copies of FAT times 251 sectors each, plus root folder 32 sectors, total 534 occupied by system data sectors).

Clusters 0 and 1 do not exist, so the first data cluster is 2.

Cluster number 3 is next to cluster 2, i.e. it is located 64 sectors behind the first data sector (535 + 64 = 599).

Equal offset of 306,668 byte from the beginning of the drive (0x4AE00).

With a help of low-level disk editor on the disk we can see our data starting with offset 0x4AE00, or cluster 3, or sector 599:

```
Offset    0  1  2  3  4  5  6  7     8  9  A  B  C  D  E  F
0004AE00 47 55 49 20 6D 6F 64 65    20 53 65 74 75 70 20 68   GUI mode Setup h
0004AE10 61 73 20 73 74 61 72 74    65 64 2E 0D 0A 43 3A 5C   as started...C:\
0004AE20 57 49 4E 4E 54 5C 44 72    69 76 65 72 20 43 61 63   WINNT\Driver Cac
```

Because the cluster chain is consecutive, all we need to do is copy 112,435 bytes starting from this place. If the cluster chain was not consecutive, we would need to re-calculate the offset for each cluster and copy 3 times the value of 64*512 = 32768 bytes starting from each cluster offset. The last cluster copy remainder, 14,131 bytes is calculated as 112,435 bytes - (3 * 32,768 bytes).

Recovering Cluster Chain in NTFS

In our example we just need to pick up 110 clusters starting from the cluster 312555.

Cluster size is 512 byte, so the offset of the first cluster would be 512 * 312555 = 160028160 = 0x0989D600

```
Offset    0  1  2  3  4  5  6  7     8  9  A  B  C  D  E  F

0989D600 D0 CF 11 E0 A1 B1 1A E1    00 00 00 00 00 00 00 00   ÐÏ.à¡±.á........
0989D610 00 00 00 00 00 00 00 00    3E 00 03 00 FE FF 09 00   ........>...þÿ..
0989D620 06 00 00 00 00 00 00 00    00 00 00 00 01 00 00 00   ................
0989D630 69 00 00 00 00 00 00 00    00 10 00 00 6B 00 00 00   i...........k...
0989D640 01 00 00 00 FE FF FF FF    00 00 00 00 6A 00 00 00   ....þÿÿÿ....j...
0989D650 FF FF FF FF FF FF FF FF    FF FF FF FF FF FF FF FF   ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
```

In the above data, data recovery is complete when data has been read from this point through 110 clusters (56320 bytes). This data is copied to another location.

5 Description of Undelete Functions

This chapter describes how to perform various functions using **Active@ UNDELETE**.

Scanning a Drive

In the left pane of the Active@ UNDELETE main screen:

Click on the drive and then on the folder that you expect may contain deleted files or folders.

If you want to re-scan a drive or a folder, select it, then click the **Refresh** toolbar button or right-click the folder and click **Refresh** from the context menu.

Searching an Unknown Drive Location

Before launching this procedure, check the **Recycle Bin** to be sure that the deleted file or folder is not there. If it is, use the standard Windows **Restore** command to recover it.

If the former location of the deleted file or folder is not known, filter out deleted files and folders on the drive first, using these steps:

1. In the Active@ UNDELETE main screen, click a drive listed in the left pane. The scan process begins.
2. After scanning is complete, all files and folders will be listed on the right-hand pane.
3. In the **Action** drop-down menu, click **Set Filter**. Alternately, right-click a drive or folder and click **Set Filter** from the context menu. The **Filter Files** screen appears.
4. Define a search pattern in the **Filter Files** dialog. for example type ***.txt** to find all files with a **txt** extension.
5. Enable the **Deleted Only** checkbox to hide non-deleted files and folders.

   By default, the filter pattern is not case sensitive. To make it case sensitive, check **Match Case** option.
6. Click **Find**. The scan process will start.
7. After the search is complete, open folders in the left pane. Files contained in these folders will be displayed in the right pane.

(i) *Note: This search pattern is the same pattern used when searching in Windows. Standard global search symbols may be used, e.g. * in pattern means that at this place can be zero or any number of any symbols.*

8. To restore deleted files, continue with instructions in the next section.

Restoring Deleted Files and Folders

Find and select the deleted file in the Active@ UNDELETE console as described above. To restore contents of the file, execute the **Undelete** command using one of the following methods:

Undelete Shield Icon

1 Click the **Undelete** shield icon on the toolbar or right-click the file, and then click **Undelete** from the context menu. The **Browse for Folder** screen appears.

2 Select a folder on another logical or physical drive, or create a new folder for recovery. Once selected a **Command Confirmation** screen appears.

3 Click **Yes** to restore the file. A **Command Complete** screen appears with a summary report naming the location of the recovered file.

4 After the recovery process is complete, make sure that the file has been restored correctly by verifying its contents.

Drag-and-Drop / Copy-and-Paste

This method does not include the option to create a new folder. If you wish to create a new folder for recovery, use the previous method.

1 In the left pane, locate a folder on another logical or physical drive for recovery. Expand folders using the plus sign (+) to locate a folder. If you click the actual folder, the report in the right pane will change to display the contents of the current folder.

2 In the right pane, click the file to be recovered.

3 Drag it to the selected recovery folder and drop it there or right-click the file and click **Copy** from the context menu. Right-click the recovery folder and click **Paste** from the context menu.

4 After the recovery process is complete, make sure that the file has been restored correctly by verifying its contents.

(i) *Note: In some cases, a file can not be reliably restored, because its contents or a part of it has already been overwritten*

Restoring a Deleted Folder

Find and select a deleted folder to be restored in the Active@ UNDELETE console. A deleted folder will appear in the left pane or the right pane, depending on whether a logical drive or a sub-folder is selected. Clicking a deleted folder in the left pane will reveal nothing in the right pane.

To recursively restore the contents of the folder with files and subfolders, the process is the same as described in the previous section about restoring files. Run the **Undelete** command by one of the following methods:

1 Click the folder in the left or right pane. Click the **Undelete** shield icon on the toolbar.

2 Right-click the folder, then click **Undelete** from the context menu.

3 Drag and Drop the selected folder to another logical drive.

4 **Copy** and **Paste** the contents of the folder, using the Microsoft Windows clipboard (**Copy** command in the context menu, **Copy** button on the toolbar or **Copy** command under the **Action** drop-down menu) and paste file to another logical drive (Paste command in similar locations).

5 After the recovery process is complete, make sure that the results are correct by verifying the contents of the files and subfolders.

Undelete Wizard for File Recovery

Advanced file recovery features can be helpful if standard procedures are not successful. If successful, the Undelete Wizard can help recover a file that has been partially overwritten with the ability to preview file contents and manipulate file cluster chain contents.

Being able to see and adjust contents of clusters composing the file will help recovering only files in a readable format, such as ***.txt, *.log, *.rtf ...**

Follow these steps to use the Undelete Wizard:

1 From the Active@ UNDELETE main screen, find and select a deleted file

2 Start the **Undelete Wizard** by

  a Clicking the Undelete Wizard icon on the toolbar.

  b Right-clicking the file and clicking **Undelete Wizard** on the context menu.

3 Read the brief procedure description on the Undelete Wizard **Welcome** screen and clear the **Show this dialog next time** checkbox if you do not want to see welcome screen next time.

4 Click **Next** to follow the wizard steps.

5 The Undelete Wizard steps you through read-only **File Information**, the interactive **File Composer** screen, and the **Finish** screen.

In the **File Composer** screen, Undelete Wizard chooses a cluster chain for you. On the left is a list of all available clusters. Clusters occupied by other files are colored black Unoccupied, or free clusters are colored red. Clusters that appear grey are those that have been selected for this recovery.

At the right side there is a list of clusters composing file body. These clusters will be recovered after the wizard **Finish** screen. Click any cluster in either listbox to display its contents in the **Preview** pane below. Image buttons help manipulate the order of the clusters in the chain.

6 When all output parameters have been defined, click Finish to complete the recovery process.

Creating a Disk Image

A **Disk Image** is a mirror of the contents of a logical drive that is stored in a single file. It can be useful to back up the contents of an entire drive, in order to be able to work with it later. Before starting recovery of deleted files, it might be a good idea to create a Disk Image, if you have enough space. If something goes wrong while recovering files, the original deleted file state will be preserved in the Disk Image.

Follow these steps to create a Disk Image:

1 In the Active@ UNDELETE main screen, select a drive as an image source.

2 Click the **Create Disk Image** button on the toolbar or right-click the selected drive, and click the **Create Disk Image** command on the context menu.

3 Select the Disk Image location in the **File Save** dialog and click **Save.**

4 Watch the creation progress and wait while drive's contents are copied to the new location.

You can cancel the process of image creation if necessary anytime by clicking **Cancel**.

(!) NOTE: The Target Location for the ***Create Disk Image*** *command must always be specified on other drive.*

Opening Disk Image

Disk Image is a mirror of your logical drive that is stored in a single file. To open the Disk Image follow these steps:

1 In the left pane of the Active@ UNDELETE main screen, select the **Active UNDELETE** node or any drive node.

2 Click the **Open Disk Image** button on Snapin's toolbar or right-click a drive or the **Active UNDELETE** node, and click **Open Disk Image** on the context menu.

3 Click an existing Disk Image (file with **DIM** extension) in **File Open** dialog.

4 Click **Open**.

5 Work with an opened Disk Image the same way as with a regular drive to **Scan**, **Find** and **Restore** files from it.

Registering Active@ UNDELETE

Follow these steps to register the product:

1 If opening the software for the first time, this dialog appears automatically. Otherwise, in the Active@ UNDELETE main screen, right-click the **Active UNDELETE** node, click select **Register** from the context menu. The Registration dialog appears.

2 Enter the following information:

a Type your name into the Name field.

(i) NOTE: The spelling of your name should be exactly the same as it was specified when you purchased the product.

b The **Registration Key** is sent to you by email after you have purchased the product. **Copy** and **Paste** the product registration key into the **Registration Key** area.

(!) NOTE: Please do not try to type ***Registration Key****, just use Copy and Paste standard Windows operations.*

c Read and understand the **License Agreement**, and if you agree to the license terms, click **Apply** or **Ok**.

3 If either the name or registration key, or both are misspelled, an error message appears and the product will not be registered. If this message appears, correct registration information and try again.

If no error messages appear, the product has been registered successfully.

Configuring Preferences

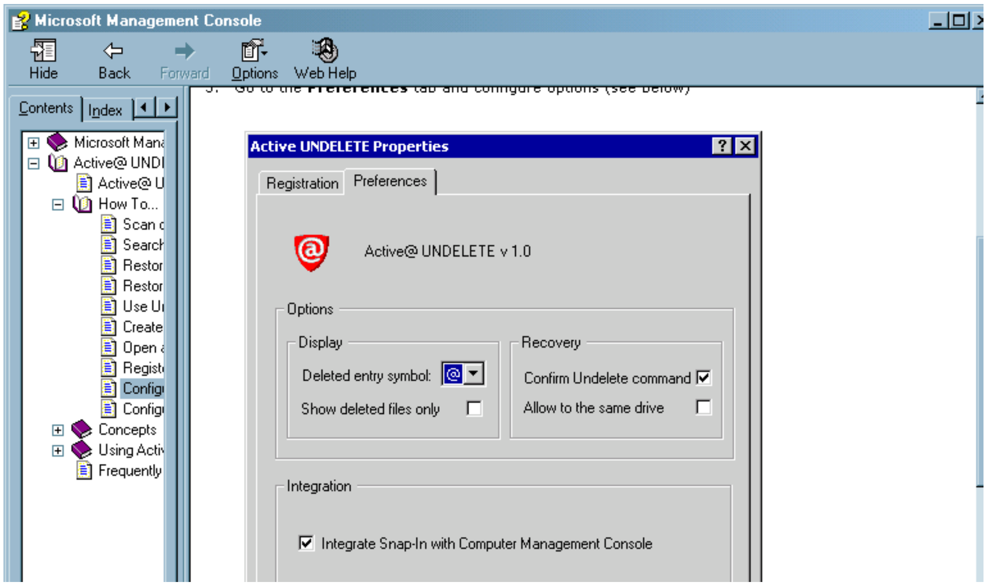
Active@ UNDELETE allows changes in local preferences for information display and recovery settings. To change local preferences:

1 In the Active@ UNDELETE main screen, click the **Active UNDELETE** node.

2 Click the **Properties** button on the toolbar or right-click the **Active UNDELETE** node and click **Properties** from the context menu.

3 Click the **Preferences** tab and configure options from information in the table below. Click **OK** or **Apply** to use new settings.

Figure 5-1 Preferences Settings



Active@ UNDELETE v 1.0

Options

Display

Deleted entry symbol: @

Show deleted files only □

Recovery

Confirm Undelete command ☑

Allow to the same drive □

Integration

☑ Integrate Snap-In with Computer Management Console

Table 5-1 Display Options

Option Name	Default	Alternates
Display		
Deleted entry symbol	ASCII#229 (@). This symbol is recognized by the operating system on FAT volumes to mark deleted file or folder.	Choose another ASCII symbol from the list: · Dollar sign ($) · Exclamation mark (!) · Number sign (#))
Show deleted files only	Disabled	Enable this check box to display only deleted files. Note: All folders will be displayed. Non-deleted files will not be displayed.
Recovery		
Confirm unerase command	Enabled	Clear this checkbox to eliminate the confirm dialog.

Option Name	Default	Alternates
Allow the same drive	Disabled	Enable this checkbox only if you encounter restricted drive space (hard drives, floppies, network, Zip...). When enabled, it will allow recovery of a deleted file on the same drive where it was located before deletion. But be careful, use this opportunity only if there are no other options. Deleted data can be damaged while recovery is in progress (see note below).
Integration		
Integrate with Computer Management Console	Enabled Note: This option is available in **Active@ UNDELETE Enterprise version** only.	If your operating system supports **Computer Management Console**, you can easily integrate **Active@ UNDELETE SnapIn** by enabling this option. It will be displayed and accessible the same way, as other Management Console features, like **Event Viewer**. See Network Configuration for more information.

(!) *NOTE: WE STRONGLY RECOMMEND THAT YOU TO SAVE RECOVERED DATA ONTO A DIFFERENT DRIVE FROM THAT WHERE YOU FOUND DELETED FILES! Saving recovered data onto the same drive where sensitive data was located, can overwrite table records for this and other deleted entries. It is safer to save data onto another logical, removable, network or floppy drive.*

Configuring a Network Installation

The **User Defined Console** allows you to configure **MMC Snap-In** to access a remote computer, see its drives and undelete files remotely. Change the connected computer for the current Snap-In, or add another Snap-In to the current MMC Console.

(i) *NOTE: Network configuration is available for* ***Enterprise version*** *only.*

To change the connected computer:

1. In the Active@ UNDELETE main screen, click the **Active UNDELETE** node.
2. Click the **Properties** button on the toolbar or right-click the Active UNDELETE node, and click **Properties** from the context menu.
3. Click the **Network** tab. Set configuration options according to the table below.
4. Click the **Snap-In** node that you configured and click **Refresh** to connect to the remote computer.
5. If there is a configuration problem, error messages will appear (see **Important** section below). Go to the **Snap-Ins Properties** screen to make changes to **Network** settings.

To add another Snap-In to the current MMC Console:

1 In the Active@ UNDELETE main screen, click **Console - New**.

2 Press **[Ctrl+M]** on the keyboard, or click **Add/Remove Snap-in...** from the **Console** drop-down menu.

3 Click **Add...**

4 Click **Active@ UNDELETE** from the list of available **Snap-Ins**.

5 Click **Add...** once more. The **Active@ UNDELETE Configuration Wizard** screen appears.

6 At the **Registration** dialog enter, or verify that registration information is correct, and click **Next**.

7 In the **Network** dialog, enter configuration options (see below) to access a remote computer.

8 When complete, click **Finish**. the Add New Snap-In screen appears again.

9 Click **Close** to return to the Add/Remove Snap-In screen.

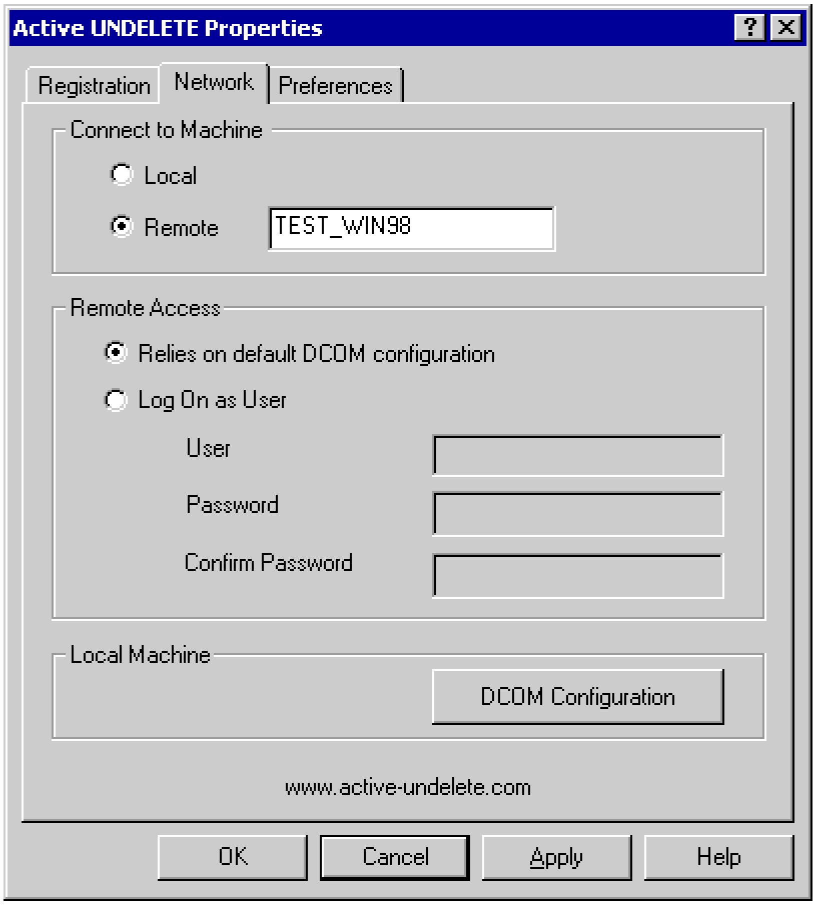
10 Click **OK** to complete the process.

11 Click the newly-created Snap-In node to connect to and work with remote computer.

12 If there is a configuration problem, error messages will appear (see **Important** section below). Go to the **Snap-Ins Properties** screen to make changes to **Network** settings.

After completing the new Snap-In configuration, save the **Console** for future use.

Figure 5-2 Network Properties



Connect to Machine

○ Local

● Remote TEST_WIN98

Remote Access

● Relies on default DCOM configuration

○ Log On as User

User

Password

Confirm Password

Local Machine

Table 5-2 Network Configuration Options

Option Name	Default	Alternates
Connect to Machine	Local Connect to and see drives on a local machine	Remote Enter the network name of the machine to connect to, and after **Refresh** command, see remote drives
Remote Access	Relies on DCOM configuration No explicit account is specified. Everything depends on how remote COM server is configured	Log On as User Specify the account, which will be used to connect to remote instances of Active@ UNDELETE: Enter user name and password used to connect to the remote server.
Local Machine DCOM Configuration	Run the standard Windows utility **DCOMCNFG.EXE** to specify how each instance of Active@ UNDELETE on the local computer will be accessible by other users in the network. See Understanding Distributed COM (DCOM) for more information.	

Computer Management Console

Another way to connect and work with remote computer is using standard **Computer Management Console**. Use this method only if operating systems on both local and remote computer support **Computer Management** (for example both machines have Windows 2000 installed).

To connect and manage remote computers:

1. Right-click the **My Computer** desktop icon and click **Manage** command or open Microsoft **Control Panel** and click **Computer Management** from the **Administrative Tools** folder.
2. Right-click the Computer Management node and click **Connect to another computer...** menu item or select it from the **Action** menu.
3. Select one of available computers, or type its name in the **Select Computer** dialog and click **Ok**.
4. The **Active@ UNDELETE** icon appears under **System Tools** node in **Computer Management**. Expand the Active@ UNDELETE node and work with remote computer the same way as with a local machine.

Important

With Active@ UNDELETE installed and registered on a remote computer, and correct access to the remote computer and DCOM is configured properly, all Active@ UNDELETE functions will run the same way as if it was a local machine.

Table 5-3 Error Messages

Error	Description
The RPC Server is not Available	Mistyped remote computer name, or software is not installed properly
Access denied	Incorrect privileges on the remote computer
Access Denied DCOM security settings do not allow this process to get notifications from the remote server	DCOM security is turned on by default. Turn it off.

See Understanding Distributed COM (DCOM) in Chapter 3 for more information on how to configure DCOM.

For testing purposes, you can try to turn off DCOM security temporarily, by setting on both computers **Default Authentication Level** to (**None**), and adding **Everyone** to **Default Access and Launch** permissions.

For more information on how to create and configure MMC Consoles, Add/Remove Snap-Ins see Understanding Microsoft Management Console (MMC) in Chapter 3.

6 COMMON QUESTIONS

The following tips are designed to help with data recovery.

Q: How do I download the trial version of Active@ UNDELETE utility?

A: You can download it from our web-site (http://www.active-undelete.com). The trial version is a utility with full functionality of the final program. The only limitation is the maximum size of the file being restored.

Q: I cannot see my drive in the list of drives. What to do if my partition is damaged or deleted? Are there other tools I can use to save my data?

A: Active@ UNDELETE works with only existing drives. It cannot access damaged or deleted partitions.

If your partition is damaged or deleted (Active@ UNDELETE screen shows the drive is not visible in the list of available drives), please try one of these other quality products from **Active Data Recovery Software**:

Active@ Partition Recovery for DOS (http://www.partition-recovery.com) **-** Can help recover damaged or deleted partition structure in a DOS environment.

Active@ UNERASER for DOS (http://www.uneraser.com) **-** Can help unerase or copy files and folders from an existing, deleted or damaged partition onto another safe drive from within a DOS environment.

Q: I have deleted a very important document. It was deleted BEFORE Active@ UNDELETE was installed on my computer. Is it possible to restore it?

A: Yes, if the file has not already been over-written (by another file).

As soon as you discover that an important file is deleted, download and install Active@ UNDELETE and search for this file. It is a good idea to avoid disk activity on the particular hard drive where the file should have been. Here are some tips:

- Do not add or delete files or applications on this drive
- Avoid restarting the computer, if possible
- Do not invoke a large number of programs concurrently (this causes increased page-swapping activity)

Any of these activities might overwrite or partially overwrite a deleted document. Any changes to the Files Table will make finding a deleted file more complicated.

The more free hard drive space you have on your computer, the greater the chances for a successful retrieval of deleted file contents. It is always a good idea to extract and install Active@ UNDELETE to a SEPARATE physical or logical hard drive - one that does not contain important deleted files.

Q: Does Active@ UNDELETE work under Windows 2000 / XP?

A: Yes, it does.

Q: Does Active@ UNDELETE work under Windows 3.x?

A: No, support of 16-bit operation systems like Windows 3.1 is not implemented.

Q: I have Netscape Navigator 4.6 as my default browser. Will I be able to install and use Active@ UNDELETE?

A: Yes. The function of downloading and installing software requires **Internet Explorer** or **Netscape Navigator**, or any other browser that supports the file download function. After software has been downloaded, the browser is no longer required to install or run the application.

Q: Does Active@ UNDELETE support localized (e.g. French, Spanish) files names?

A: Yes, provided the OS and file system support localized file names.

Q: Will Active@ UNDELETE recover long file names?

A: Yes, provided the OS and file system support long file names.

Q: What is a Disk Image? Why is it needed?

A: A **Disk Image** is a mirror of the contents of a logical drive that is stored in a single file. It can be useful to back up the contents of an entire drive, to be able to work with it later.

Before starting recovery of deleted files, it might be a good idea to create a Disk Image, if you have enough space. If something goes wrong while recovering files, the original deleted file state will be preserved in the Disk Image.

Active Data Recovery Software

Active Data Recovery Software is a software development company designing disk utilities related to the recovery of lost data and online privacy. Unique Active@ technologies allow our solutions to be easily integrated with operating system and network environment, suggesting to user powerful and flexible tools for computer management.

Contact Information
P.O. Box 13527
3221 Derry Road West #3
Mississauga, Ontario
Canada   L5N 8G5
Phone : (416) 333-5422
Customer Service: sales@active-undelete.com
Technical Support: support@active-undelete.com

http://www.active-undelete.com