

Table of Contents

CnW Recovery	6
Introduction	10
Demo program	14
Demo Status	15
Software limitations	16
Installation	17
Dongle installation	19
Media detection	22
Configuration for CnW Recovery Software	26
Recovery options	28
Registration	30
Directories	32
Hardware config	34
Basic Rules of Data Recovery	35
What to do when media has failed	35
Hardware failure - what next?	37
Welcome screen and wizard	39
Unrecognised media	42
Mini DVD recovery	43
Create new video DVD	45
Photo Recovery	46
AVCHD recovery	47
3GP and MP4 recovery wizard	49
ZIP and DOCX recovery wizard	56
Video scan of hard drive	58
Verify disk structure	59
Physical Media Test	62
Failing disk drive	64
Formatted disk recovery	65
Partitioned disk recovery	67
AVI Recovery	68
Forensic Data recovery	69
Recovering Files	70
Getting started - General data recovery	70
Typical data recovery procedures	73
View sector on hard drive, flash memory or CD	74
Disk imaging	76
Raid disks	78
How to use incremental imaging to recover damaged drives	80
Disks with single head failure	82
Image file selection	83
Partitions, analysis and recovery	85
Partition analysis mode	88
Partition Table structure	90
How to recover corrupted partitions	92
GUID Partition tables	94
Magnetic Media Recognition	95
Deleted file recovery	96
CD Recovery	98
How to recognise type of CD/DVD	100

UDF Anchor Volume	103
Unerase CD-RW	104
Multi-session UDF	107
Camcorder Recovery	108
Rebuild video disk files	110
FAT Disk restore	112
How to recovery FAT disk when boot sector and one FAT is missing	116
BIOS Parameter FDC descriptor for FAT	118
Missing directories and files on a FAT disk	120
How to recover FAT disk when boot sector is missing	121
FAT 32 deleted file recovery	122
FAT File allocation table validation and correction	124
Recover FAT32 disk when it has been reformatted as NTFS	125
exFAT	126
Linux and Unix recovery	127
Macintosh Recovery	130
MTF .BKF files	133
NTFS Recovery	134
BIOS Parameter FDC descriptor for NTFS	140
NTFS MTF range	141
Search for MFTs	142
Files lost when NTFS reloaded	144
Cannot read first mft, copy failed	146
NTFS with confused partitions	147
Alternate Data Stream	149
Recovering when a new /different operating has been loaded	151
Deleted Partition	153
How to find and recover lost files	154
Recovery from a drive with many bad sectors	156
Data carving options	158
Raw files	161
Search String	163
Recovering files from image format	165
Fragmented file processing	167
Jpeg images and metadata	170
Fragmented Files	171
Fragmented AVI files	173
Tutorials	174
General NTFS Recovery	175
Recover video from camcorder with a hard drive	178
Recovery of lost files on an otherwise working disk	179
Photo recovery	180
Imaging failing drive	181
Video file recovery	183
Video recovery from mini-DVDs	183
Video recovery from memory devices	185
MP4 disk layouts	186
mp4_scan	190
MP4 file structure	191
GoPro video recovery	193
HP MediaVault data recovery	195
HP Mediavault Tutorial	196
Forensic Tools	200
CnW Recovery forensic investigation tools	201

Discover deleted files	204
ISO9660 and Joliet investigation	206
UDF forensic investigation	207
NTFS forensic investigation	209
MFT Parse	211
DVD Properties	213
Data Carving	215
Manual Data Carving	217
Data Carving with an Excel File	218
File Validation	221
Search for strings	222
XML Forensic Report	223
NSRL Hash tables	226
Disk scan	228
Virtual disk image	230
Forensic analysis tips	231
File selection	232
Overview	232
File extension selection	234
Date selection	235
Directory selection	236
File name selection	237
File selection based on MD5 value	238
File size selection	240
Import List	242
RAID drives	243
RAID drive selection	245
RAID configuration	247
RAID boxes and configurations	249
RAID JBOD	251
Typical RAID setup parameters	253
HP Mediavault recovery	254
Fragmented files	256
Fragmented 3GP/MP4 files	257
Typical 3GP corruptions	260
Fragmented Zip and DOCX files	261
Recognising Sectors	265
Master Boot Record	265
GUID Partition sectors	267
BIOS Parameter Block BPB	270
FAT directory entry	273
NTFS directory entry, MFT	275
Disk clusters	277
Apple Volume Header	279
VMFS sectors	281
General tools	283
Split directories	283
Merge disk images	284
eMail Extraction	285
User passwords	286
AVCHD reconstruction	287
Extract and join	288
Fake memory test	289
Reconstruction tips	290

eMail restoration	290
Logs	291
Log overview	291
File details	293
Search for sector	297
File fragments	298
Job details	299
Forensic Report	302
Keyword search	304
Trace file	306
Errors and problems	307
An error occurred in an unknown file	307
Missing files	309
Error Messages	310
All files are short	311
Files not saved	312
CD physical structures	313
Disk at once	313
Track at once	315
Session at once	316
Packet writing	317
CD terms	318
Glossary	320
Terms	320
Useful links	321
Addresses, and contact details	322

CnW Recovery Software

Manual for data recovery software for all PC storage devices

[Main Website](#)

- [Introduction](#)
 - [Demo program](#)
 - [Demo Status](#)
 - [Software limitations](#)
 - [Installation](#)
 - [Dongle installation](#)
 - [Media detection](#)
 - [Configuration of CnW Software](#)
 - [Recovery options](#)
 - [Registration](#)
 - [Directories](#)
 - [Hardware configuration](#)
- [Basic Rules of data recovery](#)
 - [What to do when media has failed](#)
 - [Hardware failure - what next?](#)
- [Welcome screen and wizard](#)
 - [3GP and MP4 recovery wizard](#)
 - [AVI Recovery](#)
 - [AVCHD recovery](#)
 - [Photo recovery](#)
 - [Mini DVD recovery](#)
 - [Create new DVD](#)
 - [Video scan of hard drive](#)
 - [Formatted disk recovery](#)
 - [Partitioned disk recovery](#)
 - [Failing disk drive](#)
 - [Physical media test](#)
 - [Verify disk structure](#)
 - [DOCX and Zip recovery](#)
 - [Unrecognised media](#)
- [Forensic Data Recovery](#)
- [Recovering files - start here](#)
 - [Getting started](#)
 - [Typical data recovery procedures](#)
 - [View sector on hard drive](#)
 - [Disk imaging](#)
 - [Raid disks](#)

- [How to use incremental imaging to recover damaged drives](#)
- [Disks with single head failure](#)
- [Deleted file recovery](#)
- [CD Recovery](#)
 - [How to recognise type of CD/DVD](#)
 - [UDF Anchor Volume](#)
 - [Unerase CD-RW](#)
 - [Multi-session UDF](#)
 -
- [Camcorder recovery](#)
 - [Rebuild video disk files](#)
- [Partitions, analysis and recovery](#)
-
- [FAT Disk recovery](#)
 - [How to recover FAT disk when boot sector and one FAT is missing](#)
 - [BIOS Parameter FDC descriptor](#)
 - [Missing directories and files on a FAT disk](#)
 - [How to recover FAT disk when boot sector missing](#)
 - [FAT32 deleted file recovery](#)
 - [FAT File allocation table validation and correction](#)
 - [Recover FAT32 disk when it has been reformatted as NTFS](#)
 -
- [exFAT recovery](#)
- [Recovering when a new operation system has been loaded](#)
- [Linux and Unix recovery](#)
- [Macintosh recovery](#)
- [MTF BKF Files](#)
- [NTFS recovery](#)
 - [BIOS parameter FDC description](#)
 - [NTFS MFT range](#)
 - [Search for MFTs](#)
 - [Cannot read first MFT, copy failed](#)
 - [Files lost when NTFS reloaded](#)
 - [NTFS with confused partitions](#)
 - [Recovering files when a new or different file system has been loaded](#)
-
- [RAID recovery](#)
 - [RAID drive selection](#)
 - [RAID configuration](#)
 - [RAID JOB](#)
 - [Typical RAID setups](#)
 - [HP Media Vault and Broadcom](#)
 -
- [MTF .BKF files and recovery](#)
- [Deleted partition](#)

- [How to find and recover lost files](#)
- [Recovery from a drive with many bad sectors](#)
- [Disk Image](#)
 - [Recovering files from image format](#)
 - [RAID 0 disks](#)
 - [How to use incremental imaging](#)
- [Data carving options](#)
 - [Raw files](#)
 - [Search String](#)
- [Tutorials](#)
 - [General NTFS recovery](#)
 - [Recover video from a camcorder with a hard drive - rather than a DVD](#)
 - [Recovery of lost files on an otherwise working disk](#)
 - [Photo recovery](#)
 - [Image a failing drive](#)
- [HP Mediavault data recovery](#)
 - [HP Mediavault](#)
- [RAID drives](#)
 - [RAID drive selection](#)
 - [RAID configuration](#)
 - [RAID boxes and configurations](#)
 - [RAID JBOD](#)
 - [Typical RAID setup parameters](#)
 - [HP Mediavault recovery from a RAID](#)
- [Video recovery](#)
 - [Video recovery from mini-DVD](#)
 - [Video recovery from memory chips](#)
 - [MP4 disk layouts](#)
- [Basic Rules of Recovery](#)
 - [What to do when media has failed](#)
 - [Hardware failure, what next](#)
- [Forensic tools](#)
 - [CnW Recovery forensic investigation tools](#)
 - [Discover deleted files](#)
 - [ISO9660 and Joliet investigation](#)
 - [UDF forensic investigation](#)
 - [NTFS forensic investigation](#)
 - [MFT parse](#)
 - [DVD properties](#)
 - [Manual data carving](#)

- [File validation](#)
- [Forensic analysis tips](#)
- [Forensic Data Recovery](#)
- [Recovering Files](#)
 - [Getting started - General data recovery](#)
 - [Typical data recovery procedures](#)
 - [View sector on hard drive](#)
 - [Disk imaging](#)
- [File Selection](#)
 - [Overview](#)
 - [File name selection](#)
 - [Date selection](#)
 - [Directory selection](#)
 - [File selection based on MD5 value](#)
 - [File size selection](#)
 - [Import List](#)
- [Forensic Tools](#)
 - [Recovery and forensic investigation tools](#)
 - [Discover deleted files](#)
 - [ISO9660 and Joliet investigation](#)
 - [UDF forensic investigation](#)
 - [NTFS forensic investigation](#)
 - [DVD properties](#)
 - [Data carving](#)
 - [Manual data carving](#)
 - [File validation](#)
 - [Forensic Reports](#) - in XML
- [Logs](#)
 - [Log overview](#)
 - [File details](#)
 - [Search for sector](#)
 - [File fragments](#)
 - [Job details](#)
 - [Forensic report](#)
 - [Keyword search](#)

Last updated June 2022
Copyright CnW Recovery Developments Ltd

-0-

Introduction CnW Data Recovery Software

[Home](#)

Recover lost or deleted data, files, photos and videos
Partially failed, corrupted or quick reformatted disks
[Forensic](#) investigation tools

CnW Recovery is a program to recover and restore data from all CDs, DVDs, hard drives and memory chips. It recovers data from damaged and corrupted disks, memory chips, CDs and DVDs. It optionally provides a large amount of forensic analysis of the media including partial files, deleted files as well as data stored in unallocated space.

The comprehensive program is configured with many options ranging from basic data recovery of CDs, and camera memory chips, up to full and automatic forensic analysis of most media. Full logs are created for the forensic applications, along with data hashing for security to verify that data has not been changed since restored.

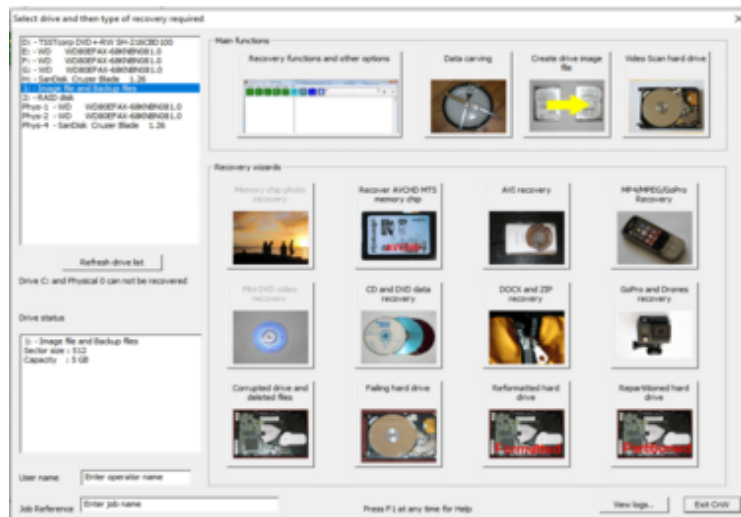


The basic functions of CnW are to list the disk properties, and to recover the data in one of several possible ways. Where ever possible, the program will determine the best mode of operation, but this can normally be overridden where the user has determined that a different mode may be more appropriate. CnW can be used to recover data from almost any Windows compatible storage device including the following

- Hard disk drives
- RAID 0 and RAID 5* configuration
- LBA48 compatible, ie disks/RAIDs larger than 2TB can be read

- Some encrypted drives
- Floppy disks
- CDs and CD-RW
- DVD and DVD-RW, including mini DVD
- Iomega Rev drives
- Optical disks
- GoPro and DJI drone support
- Flash memory
- Camera memory
- Thumb drives and Pen drives
- Jaz drives 1GB 2GB
- Zip drives 100MB, 200MB, 750MB
- Disk Image files

The program operates with USB, IDE, Firewire, S-ATA or SCSI drives on a Windows 2000, Windows XP or Vista system. Image files (dd format) may also be generated and subsequently used rather than the original media.



Logically it will read disks and media formatted as [FAT12](#), [FAT16](#), [FAT32](#), [NTFS](#), [MAC HFS Plus](#), [ISO9600](#), [Joliet](#), [UDF](#), [MTF \(.BKF files\)](#). Some Unix formats including [XFS](#), [Reiser FS](#), [HP Media Vault](#) are supported. It supports multiple partition drives, and multi-session CDs/ DVDs. Each type of media, and how to recover from it, is described in detail in the relevant section of the [Recovery](#) chapter in this manual.

Typical features that will be used in recovery include the following

- Recovery when boot sector missing or corrupted
- Recovery when disk has been re-partitioned
- Recovery after a quick format
- Wizard operation
- MP4 recovery
- AVCHD recovery
- Deleted file recovery - including intelligent recovery on FAT32 disks

- Disks with failed sectors
- One step video disk recovery and creation of new video compatible files
- Data carving
- Fragmented photo and avi recovery
- Disk imaging - included incremental images
- Viewing of sectors
- Deduplication of files
- Raw file recovery - when no file system remains
- Logging of file details - enhanced in forensic mode
- Alternate Data Stream on NTFS disks, and Mac Resource forks
- Slack* and unallocated space recovery
- Processing and recovery of fragmented files, eg JPEGs, 3GP
- Manual* data carving

(* some of the above are forensic/RAID only features).

A free [demo](#) download may be evaluated to determine if your data can be recovered. The program may then be purchased online from https://www.cnwrecovery.com/html/purchase_now.html

To help use the program, go to the [getting started tutorial](#). Amongst other points, it explains how to recover files on a corrupted hard disk.

[Typical recovery procedures](#) is a list of examples of how different types of failure or corruption can be handled.

If problems are encountered, the section on [Errors and Problems](#) may provide assistance.

With every screen of CnW Recovery software, if you press F1, there will be a context sensitive help screen. On many screens by move the mouse over a button, a small help message (tool tip) will be displayed

On many of the recovery option boxes, by double clicking on a sector value, the sector will be displayed.

CnW Recovery may be contacted by e-mail with any questions info@cnwrecovery.com or looking on the website <https://www.cnwrecovery.com>

A PDF copy of this help may be downloaded from www.cnwrecovery.com/html/cnw.pdf

Overall, the program is simple to use, and powerful enough to recover data in all instances.

-0-

Demo program

[Home](#)

The demo program is exactly the same as the production version, with the exception that files may not be restored to the hard disk. This is an important point because if the demo can not see a disk drive, or files required, neither will the licenced version.

On startup, the Welcome screen will be displayed, and once a registration code has been received, there is a function button to enter the key value. In order to complete any registration, the 8 byte Machine ID code must be sent to CnW, which will then enable the software on current machine. There is also a direct link to the online Purchase Now page of the CnW Recovery Web site. Purchase of a licence may be made using PayPal and the registration key will be returned within a few minutes, fully automatically.

Will the program recover my files?

To get the best value from the demo, it is worth trying to recovery data from a problem disk. Any files that could be recovered can be displayed, either as a Hex dump, or in the case of pictures, a small image can be displayed. The [log](#) lists all such files, and is used to view them - even on the demo.

The easiest way to recover data is by using the [Wizard](#) but for more complex setups, the Main [Recovery functions and other options](#) will be required.

Along with recovery tools, CnW Recovery has useful tools such as sector viewing and partition analysis, and extensive logging features.

Options

The standard demo does not enable the forensic features or RAID features. If these need to be evaluated, please contact CnW for details of how this option can be viewed in demo mode.

Support questions

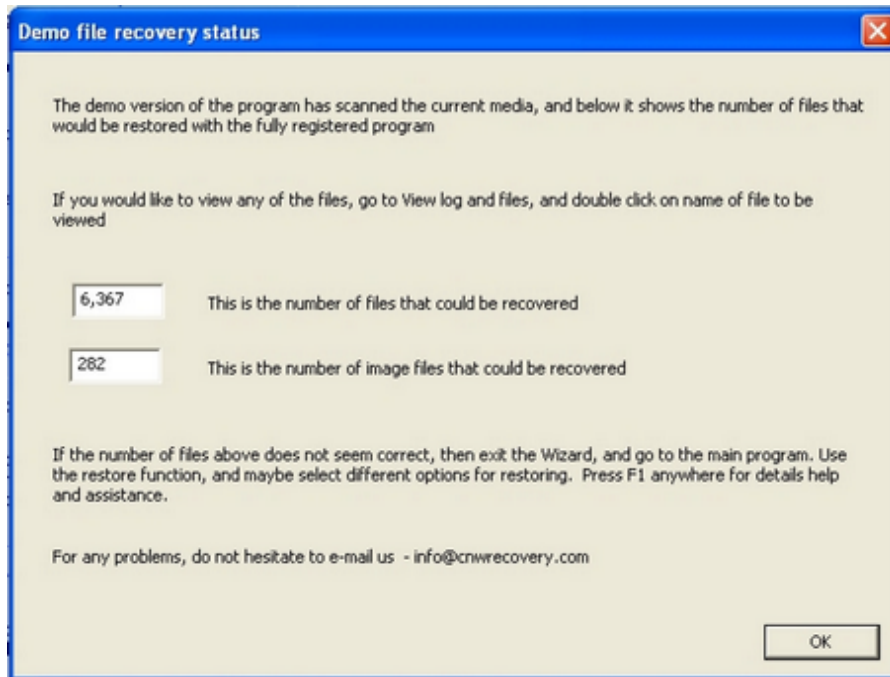
Any query, or question will be responded to typically within a day, and often within the hour - depending on time zones across the world.

-o-

Demo Status

[Home](#)

When the wizard does a media scan with the Demo system, it will display a very brief screen giving details of the number of files restored, and how many were images. This screen is shown below.



This is a brief indication of what could be recovered with the fully licenced copy of the software. In order to view some of the files, it is possible on the analyse screen to open the Log and double click on on files. These will then be displayed, either in Hex, or an image.

-0-

Software limitations with failed hardware

[Home](#)

CnW Recovery is a software only program and so there are limitations on the types of recovery it can do. It does rely on being able to see the drive as a physical device within Windows, typically as a USB or IDE device.

Good signs

- Drive spins
- Heads move
- No clicking
- No very hot spots

Bad signs

- Drive not spinning
- Drive clicking
- Hot spots on controller board - or smoke

Any of the above normally means that a physical repair will be required. It could be new heads, controller board / firmware updates or a new motor. It is not possible to repair every drive, in particular those who have suffer platter damage.

DO NOT open a drive unless you are qualified repairer - you will do far more damage than good.

If a drive has a failure of the controller board, or will not spin then no software only solution will work. If the drive is recognised as a physical drive, then there is a very high chance that CnW Recovery will be able to recover data.

For CDs and DVDs it is always recommended that a full RW drive is used to recover data. These types of drives have features that will enable disks that have not been closed to be read. However, not every unfinalised DVD can be read - some will require services from CnW office who have a modified drive to read such disks

If there is any question of whether the program will work, the quick answer is to [download](#) the demo and try.

-0-

Installation

[Home](#)

Installation and hardware requirements

CnW data recovery software will operate on any modern PC running, Vista or Windows 7. Minimum hardware is as follows. Most machines, including laptops, less than 5 years old will be fully compatible

- 1GB RAM or better (more preferred)

- Processor 1.5GHz or better (2GHz preferred)

- 20MB disk space for program installation

- Available disk space to match the capacity of the media being restored

- Interface to media being restored – typically USB 2.0 or USB 3.0

- Version 1 is extremely slow

- USB card reader to read camera memory chips

- Interface for media to be recovered

 - For hard drives, an external USB case is good

 - For IDE laptop drives, a 3.5" to 2.5" adaptor is required, or a USB case

 - CD/DVD, a suitable RW drive, typically USB based.

- When using external drives, do ensure there is always good cooling - a drive fan is an easy solution, or a case with a large built in fan.

NEVER LOAD CnW RECOVERY SOFTWARE OR ANY OTHER RECOVERY SOFTWARE ON TO HARD DRIVE THAT HAS FAILED OR BECOME CORRUPTED. IT CAN LEAD TO MAKING A RECOVERY IMPOSSIBLE, OR LOSING MORE DATA THAN NECESSARY. Please read [What to do when media has failed](#) for more details

Vista and Windows 7, 8

With Vista and Windows 7 it is necessary to run in Administrator mode (in order to access physical drives). When the program is started, the UAC message box is displayed, and has to be accepted.

How to load the software

To load the software, run the 'CnWInstall.msi' program downloaded from www.cnwrecovery.com. The link from the download page actually downloads a small program, CnWDownload, that will then automatically download and run CnWInstall.msi. Normally the default settings will be fine. The default installation directory is

c:\Program Files\CnW Recovery\CnW

but this may be changed if required. The important files that are restored are cnw.exe and cnw.hlp. There is also a subdirectory with sample File

Selection files

If a SCSI device is to be used, then a SCSI adapter card has to be installed, and the manufacturers device drivers loaded. CnW Recovery does not use ASPI, although the program works when ASPI has been installed

How to update CnW Recovery software

The software stored on the web site (www.cnwrecovery.com/html/downloads.html) is always the current, most up to date version. The demo software, and production software are identical, but are controlled by the registration key. To upgrade the software to the latest version, download from the web, and install as above.

Hardware setup

The hardware setup can be quite varied. The, most important point to note is that CnW Recovery software works by recovering files to a different drive, rather than trying to restore files to the original drive.. The next section [Media detection](#) gives details of each type of media.

Windows XP support

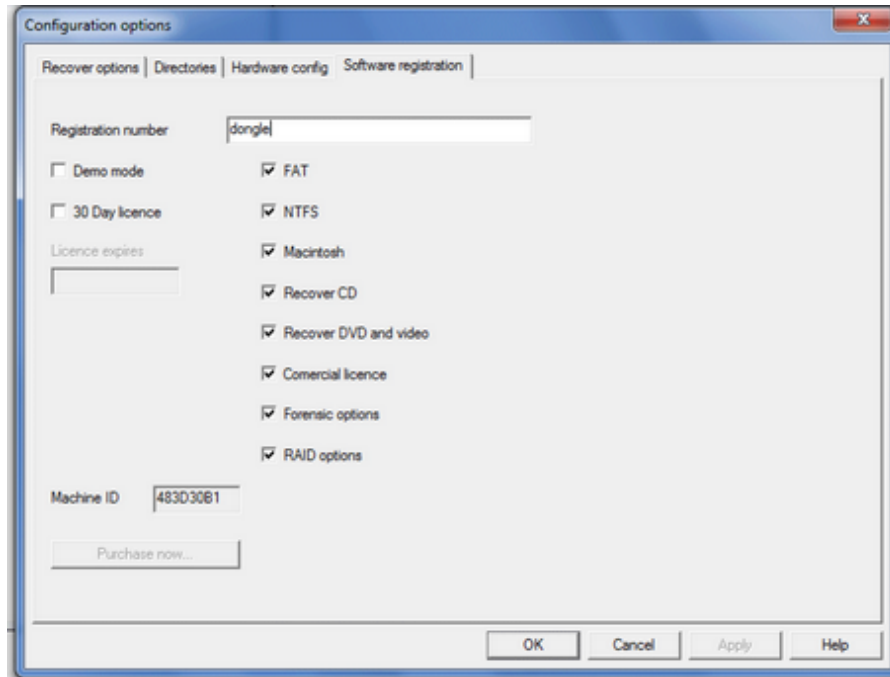
Support for Windows XP was stopped in March 2014. Any downloads on an XP system after this date will have a frozen version V4.44 installed. The reason for stopping support is to make use of Windows Media Foundation and many useful tools to assist with video recovery. The timing also coincides with Microsoft's end of support for XP

-0-

Dongle installation

[Home](#)

When a dongle is used, the registration number (in the Configuration menu) is set to "Dongle" (upper or lower case) and the dongle must be inserted in a USB port.



When first inserted, the device drivers must be loaded. This is a simple operation, as described below. It does vary between operating systems, but the following is a general guide line.

There are two current dongles, older ones are black, current ones are blue. The older black dongle was released pre Windows 8 and the drivers have not been fully updated to be signed. This can mean complex installation on Windows 8 and 10. All new dongles being sold are blue./

Windows - 10 and Blue dongle

Recent tests with Windows 10 indicate that the dongle is now detected automatically, and no additional drivers are required. The device (looking with Windows Device manager) is seen under

- Universal Serial Bus controllers
- USB Serial converter

Looking at details, the manufacturer is FTDI, and driver provider is FTDI

Windows 7-8

Driver downloads. A separate download is required for the dongle drivers. Please go to the following link [link to download](#) the drivers. The link is near the bottom of the page and the downloaded file is CnWDongle.msi This is a standard Microsoft Install file and will by default save the drivers in the program files(x86) CnW recovery\CnW directory. Drivers will not need updating between versions of CnW.

Insert the dongle in the USB port and Windows will detect new hardware. If it cannot find the driver, or give you a choice of where to search then

Go to Control Panel, System, Device manager, or device drivers

In this column, you should see the DL-D device with a yellow question mark, indicating no driver found

Select the device and the Update driver

In all cases ...

On the first menu do not search the internet for drivers

On the second menu, select the option to install from a specific location

The location to install from is in the program files, where CnW has been loaded. By default this will be

c:\program files\Cnw Recovery\cnw\CDM20802 WHQL

Certified

or

c:\program files (x86)\Cnw Recovery\cnw\CDM20802 WHQL

Certified

The driver is not certified, so please accept the next dialog warning box. Do not be put off by the red colour.

Installation will then take place. It is not necessary to reboot the PC after installation.

The installation is only required the first time the dongle is loaded on a new PC.

If the software is run without the dongle being installed, it will only operate in demo mode, ie no files will be saved.

The dongle is not a CnW produced device and the following link is from the manufacturer which can be used if there are problems with the driver installation try the www.ftdichip.com site. The dongle is based on the FT232 device, but has a non standard product ID, and so the .INF files have been modified specially to handle this device

Windows 7 64 bit compatibility. The CDM20802 version of the driver, and CnW

supplied .inf files are compatible with Windows 7-64 bit

Black dongle, Windows 8 and 10

Unfortunately the installation of Windows 8 and 10 is a bit more complex. The current driver for the dongle is not signed by the manufacturer and so a process is required to allow Windows to load an unsigned driver. This is a few simple stages, but it does require rebooting the PC, and so these instructions should be printed out to aid the process.

- 1. Windows Key + R*
- 2. Enter shutdown.exe /r /o /f /t 00*
- 3. Click the "OK" button*

System reboots here

- 4. System will restart to a "Choose an option" screen*
- 5. Select "Troubleshoot" from "Choose an option" screen*
- 6. Select "Advanced options" from "Troubleshoot" screen*
- 7. Select "Windows Startup Settings" from "Advanced options" screen*
- 8. Click "Restart" button*
- 9. System will restart to "Advanced Boot Options" screen*
- 10. Select "Disable Driver Signature Enforcement"*
- 11. Once the system starts, install the drivers as described above*

You will be asked to accept an unsigned driver. The installation of the driver does take a few minutes, so do not panic too much.

-0-

Media detection

[Home](#)

Before one can recover data, it is necessary to detect the drive and media. This section will give of how new drives and media should be attached to your PC. Each type of media is attached in a different way, but because the media has probably been corrupted, so problems may arise. Each media type is discussed in detail below.

Camera Memory Chips

A camera memory chip can not normally be read directly from a camera. There are two reasons, the first being that there are many variations of camera, and camera drivers, and the second is that very few cameras let an outside device actually access the memory. As it is often the memory that has become corrupted, it is essential to be able to access this directly, without any camera software 'in the way'.

The most satisfactory way to recover memory chips is with a dedicated, often USB card reader. These can be purchased for very reasonable sums, say \$10-20, and often read multiple types of memory chip. When used on a XP PC, Windows 7/8 PC, the memory chip will appear as a logical drive, for instance Drive H:. Once detected, CnW Recovery Software will be able to access the data and recover any files.

Memory chips can physically die so it is always worth while testing the USB - card reader etc to make sure the system will read working cards.

Hard disk drive, Zip and Jaz

There are several interfaces that may be used to access a disk drive, or memory chip. From a software stand point, it is necessary for the computer to see the drive as a physical drive, or as a logical drive, eg drive G:. The following interfaces are supported, USB, IDE, Serial-ATA, SCSI and Firewire.

USB

This is probably the most common interface to be used. A very good way to access failed hard disk drives is in a USB drive caddy. This can then be hot plugged onto a PC with no dismantling of the PC. No special drivers are required. Occasionally with USB interface, a failed drive will not be recognized immediately and it may be necessary to reboot the PC. Sometimes, going into control panel and devices and doing a hardware scan will kick the drive interface into life. There are USB adapters that will drive laptop hard drives directly – often using 2 USB connectors to get adequate power for the drive. Other versions are just a connector that converts an IDE cable to a connector compatible with a laptop drive. The biggest advantage of the USB interface is that the hot

plugging means that 9 times out of 10, it is not necessary to reboot the computer

IDE

To connect an IDE drive directly normally means partially dismantling the PC which is not normally the preferred route. Connecting a drive requires that drive ID is set correctly, ie Master or Slave ensuring that it does not conflict with other devices on the same cable.

For obvious operating reasons, CnW software will not allow analysis or recovery from the main program hard drive, ie the C: drive. Any IDE drive must be plugged in when the computer has been turned off, and should be detected on start up.

Serial ATA and SCSI

Both these interfaces either require a suitable host adapter card, or the relevant interface built into the computer motherboard. The drive may then be connected within the PC or externally. For SCSI it is not necessary to have an ASPI driver loaded, though tests show that when ASPI is loaded, the software continues to work as required. Although it may be possible to plug these devices in when the computer is running, typically it will need to be rebooted to see the new device. The alternative can be to go to the Control Panel (in Windows), select System, and then Device manager. There is an option to 'Scan for new hardware'. This will normally detect new drives, without needing to reboot the machine.

Firewire

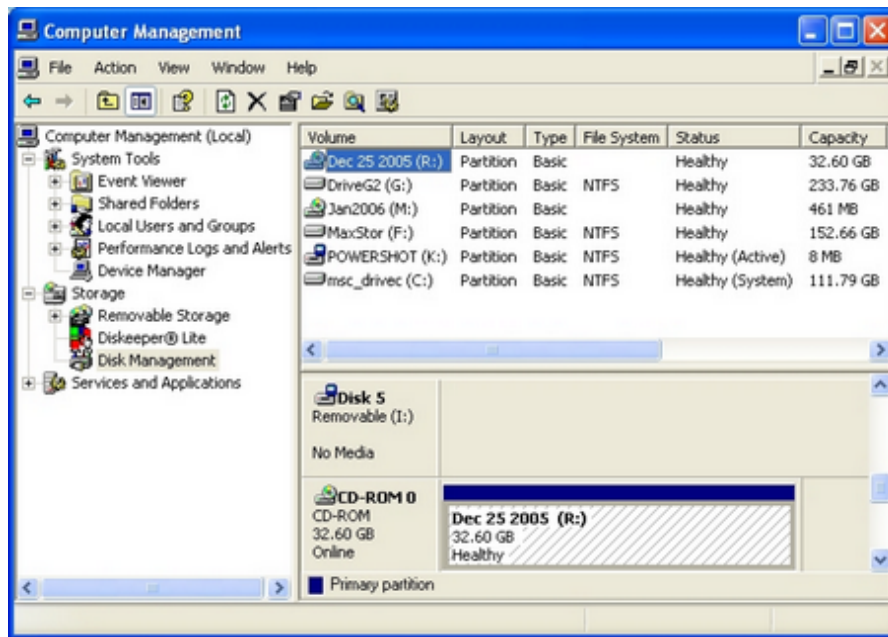
Many removable drive caddies have a Firewire interface. It can be faster than USB, but does not have the same hot plugging capability of USB. Some laptops and computers have built in Firewire interfaces, for others a special adapter board will be required. To recognise a Firewire drive, it is either necessary to reboot the computer, or use functions within the control panel (see below) to rescan for new drives. For regular swapping of drives, USB is probably easier to use.

Parallel

Parallel interfaces are normally only used on older Zip drives. Most more recent drives are USB based

Computer Management Control Panel

A very useful Windows tool is the Computer Management option within the control panel. Select Administrative Tools, and then Computer Management. The screen below will be displayed. If this program does not detect the required drive, then neither will CnW Software - the problem is likely to be hardware related, and a software solution will not help.



When you click on Disk Management, all disks that are seen by the computer are displayed. It may include unrecognised disk, unformatted etc. DO NOT TRY AND FORMAT ANY DAMAGED DISK, CnW Recovery software will do better at recovering data without an extra layer of format being added. On this screen, by Right Clicking on a disk description, it is possible to change the Drive Letter. CnW Recovery does not mind which drive letter is used, but occasionally it is possible to have a drive letter conflict with a Subst Drive.

The disk management program will sometimes show a disk as a Raw drive, or raw disk. This happens when it cannot determine the operating system, or the operating system is not a standard PC one, for instance a Macintosh disk may well show in this mode. It can also be the case when the disk has lost its boot partition. In these cases, it is often best to start by using [View Sector](#) to try and determine what type of disk it may actually be.

On the computer Management menu is an Action function. This can be used to rescan the disks, and occasionally this will add disks that have not been detected automatically.

CnW Recovery software will display drives on two ways, one as the logical drive, eg Drive :, and the other as a Physical Drive. In the screen above, the drive marked as Disk 5 would be shown with two entries, I: and Phys-5. With some disks, the system does not allocate a drive letter, and so the only way to view the drive is as a physical drive. If both modes show, selecting either one will produce identical results.

Testing

Once a drive has been connected, it is often worth doing a short test to verify the connection is correct. The simplest test is to [read a sector](#) and make sure that data can be viewed. If sector zero cannot be read, try say

100, or 0x100. For a mini DVD, try 15,000 or so, often the start of the disk is very corrupted, and unreadable.

Drive letter or disk number

CnW Recovery has two ways it will detect and represent a drive. It can either be as a logical drive, eg Drive G:, or as a physical drive, Physical Disk 3. Both are valid and will produce the same results. If a drive is not recognised by the operating system in any way, then the only option that will be displayed will be the Physical Drive number.

CD and DVD

For data recovery on a CD or DVD, a compatible drive is essential. Fortunately, most new drives are always fully backwards compatible. One important note though is that burners (ie drive that will write to a blank disk) have many more features than the simpler Read only drives. For this reason, it is always best to use a drive that is a burner.

The interface of the drive is not important. CnW Recovery software will work with all common types, such as IDE, USB and SCSI

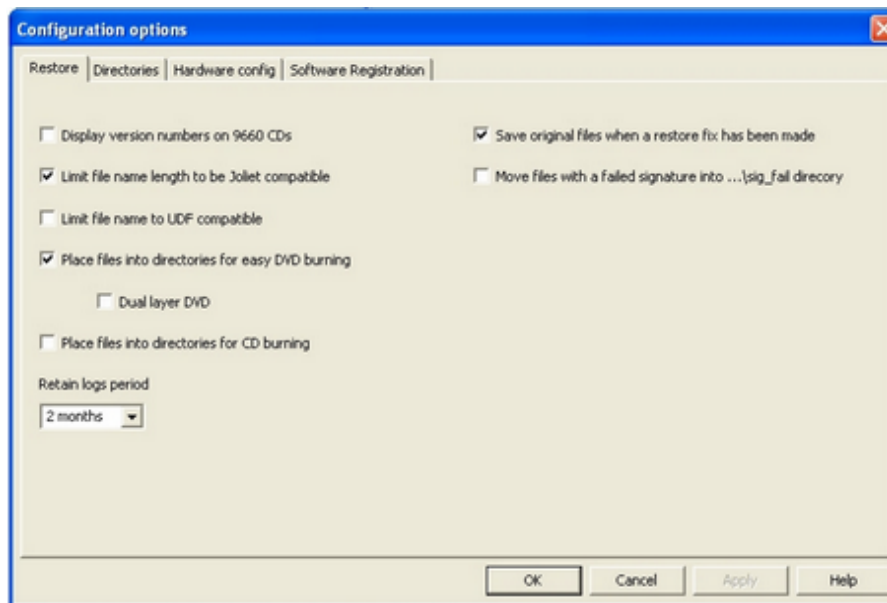
It may be noted that for some CDs and DVDs, when they are very corrupted, or unclosed etc, the startup routine of CnW Recovery software can be rather slow (minutes rather than seconds). This problem is being worked on, but patience is useful.

-0-

Configuration for CnW Recovery Software

[Home](#)

The configuration dialog boxes are for general configuration of the program. The parameters are global in their coverage and typically define how a user wants to do their data recovery. For a one off recovery, the defaults will probably be quite acceptable. The most important feature to check is if the recovery has to be written to CDs or DVDs. In this case, the Restore Options have flags that assist with file naming, and locating files into suitable subdirectories



Restore options

This range of options largely controls choices on file naming, and splitting for CDs, DVDs

Display version numbers on 9660 CDs

CDs internally append a version number to each file name. Thus a file will look like filename.ext;4 for version 4 of the file. For PCs, version numbers are not used, and so the concept is rather meaningless. By default, these version numbers are suppressed.

Limit file name length to be Joliet compatible

When writing to a CD or DVD, Joliet filenames is very common (the other usual option is UDF). Joliet filenames have a limit of 64 characters for each element of the name and path. If a CD is to be burnt or written with a longer filename, the application will normally ask to convert the names, or the user has to manually change the name. Often files that are URL addresses can be very long. This option will automatically reduce the name length to remain compatible with CD writing. The method used is first to remove any spaces within a name. If that

does not save enough space, then the middle of the name is removed, whenever possible ensuring that the file extension remains unchanged.

Limit file name to be UDF compatible

This option is much as for the Joiet above, but UDF does allow longer file name elements.

Place files in directories for easy DVD burning

The three options allow for placing files in subdirectories for burning to DVDs, Dual layer DVDs or CDs. When selected, the files are written to a subdirectory for each output media. These directories are called DV01, DV02, CD01 etc. There is some approximation of the total capacity that can be written, allowing for long files, and many short files. Typically a DVD or CD will be filled to about 90 - 95% full capacity. The option does save a considerable amount of time trying to find break points by hand.

Files that are too large for an CD or DVD (ie more than 700MB or 4GB) will be written to the selected output directory, and not included in a DV01 type subdirectory.

Save original files when a restore fix has been made

When doing a [raw read](#), some files are passed through a verify process. This may include an attempted fix on the file to try and recovery a corrupted, or incomplete file. The new file will then have a .fix extension to indicate that it has been changed. This option will allow the original files to be saved as well as the fixed version. It is recommended that this option is enabled - except when is a very severe problem with output storage.

Move files with failed signature into ...\\sig_fail directory

As part of the recovery process, files have their signature checked. If a file fails this check, it can be moved into a new directory. For a normal recovery procedure, this function has limited use as there will be many files that fail signature checks. However, for a forensic investigation that may be interested in JPGs, this function will separate out JPG files that have been renamed, in a possible attempt to hide them.

This menu is reached from the Main Functions screen. If the program has just been started, and the wizard is displayed, just select any drive, and then the Main Functions option. At this point the Configure icon will be displayed, and the options can then be selected.

Recovery options

[Home](#)

These are options of how files should be restored, and control file names, and sometimes, file locations.

- CD-ROMs (IOS9600 and Joliet) store a version number as part of the file name. Typically this number is not required when saving to a Windows system, and this option (by default) removes the extension number
- Limit file name length to be Joliet compatible. CDs and DVDs have certain limitations on file name lengths. This option will automatically ensure that all file names are compatible with writing to a CD. It obviously has to reduce the file name, and initially it will do it by removing spaces from the file name. If this is not adequate, it will start removing characters from the middle of the name, leaving the first characters and final characters unaffected. When it is necessary to reduce the length of a subdirectory, it will always start with the last subdirectory in the chain.
- Move files that fail the signature into the `...\sig_fail` directory. Most common file types have a signature that is recognizable. For files that fail this test, they can be separated into a different subdirectory for later investigation.
- Place files into directories for easy burning. This is an option that is useful when restoring from hard disks, and the output is required on DVDs (or CDs). The files are split into subdirectories under headings of DISK1, DISK2 etc. Each directory will be filled to an approx capacity of a DVD (or CD). Files which are bigger than the capacity of the output media, will be stored in a separate subdirectory for manual processing, for instance they could be Zipped. When creating these DISK directories, all subdirectories are retained. This can lead to a problem of file names getting too long – in which case they will be truncated – and so the output path should be as near the root as possible, and as short as possible. Good examples would be `F:\R1` `F:\R2`. When the DVD option is selected, files directories will be filled to about 4.7GB, for CDs, approx 650M. Dual layer DVDs can also be split.
- Save original files when a fix has been made. Certain files are scanned for integrity when restored from unallocated space. This is an option to retain an original file as well as the fixed file. By deleting the original file, at times a significant amount of disk space may be saved, but for forensic investigation, this may not be an important criteria
- Retain Logs period. Logs grow with use. This option will allow users to automatically delete logs after a period of time – by default this is 3

months.

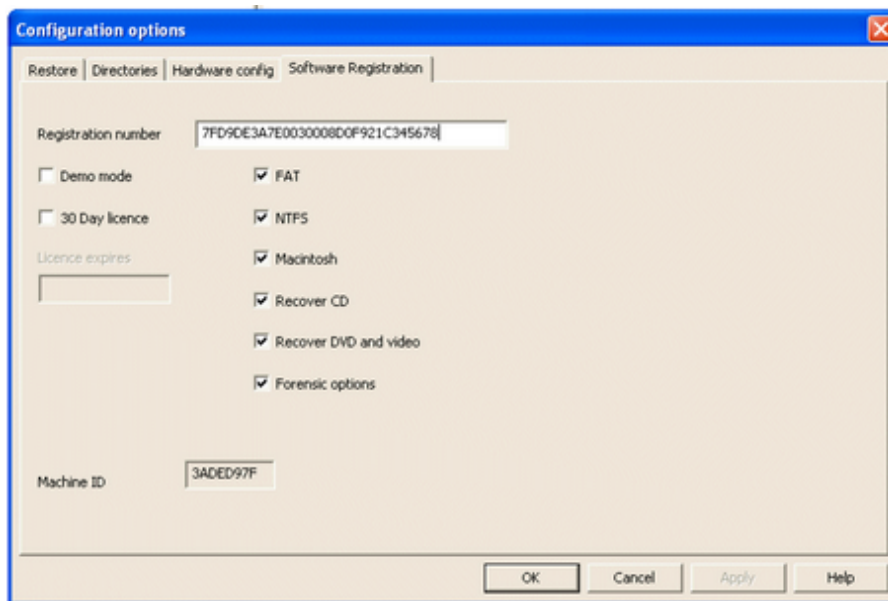
-0-

Registration and options

[Home](#)

In order for the software to be used to restore data, rather than just view it with the demo, the product must be registered (ie purchased).

Registration is controlled by the dialog box shown below that may be selected by poressing the (blue) 'Config' function on the main menu bar. There is also a short cut from the first welcome box when running in Demo mode.



The product can be purchased online www.cnwrecovery.com after it is necessary to submit the Machine ID in order to receive registration Code. The registration code is such as "7618 3200 7A00 3000 3B0C 4C96C566". This must be 'cut and paste' into the Registration Number box shown above. It is the necessary to restart the program (not the PC) before the new registration is enabled. Once restarted, it will be possible to verify the options enabled. Clicking on the check boxes will not change the selected options.

When a Dongle is used, the registration number is 'dongle'

To make entering the number by hand (cut and paste is easier though) spaces may be included between the numbers. These spaces are ignored and so are optional.

Options

CnW Software is sold with a few options, as listed below

Demo

The demo is fully functional, but does not save any files onto the hard drive. This also means that it is not possible to create a disk image file, or handle the data carving function of processing fragmented files. The demo does not have any time limitation. The demo will give a very good indication if drives and disks can be recovered. By using the log, images (normal hex dumps) of files can be viewed.

30 Day Licence

The 30 day licence is fully functional and will operate for 1 month. At the end of that period, it reverts back to the demo mode

Full licence

If both Demo and 30 day licence boxes are clear, then it is a full licence, that will work for an unlimited period of time. The program may be updated a frequently as required for the first year of purchase after which a support, update fee will be chargeable. The program will never time out, though regular updates will exist.

Raid Options

This enables the optional RAID software and can be purchased with a licence - but not part of 30 day licence.

Forensic Options

The forensic options allow further investigation of a disk. The following list below will grow, and currently includes

- Hash values for all files recovered
- [Forensic Reports](#)
- [Recovery of quick formatted CD-RW disks](#)

-0-

Directories

[Home](#)

As part of the data recovery process, the directories options allow users to store files in locations suitable for their own computer. By default, all working data is stored under a directory c:\cnwdata, but these can be moved to any other logical location or drive.

Image directory

The image directory is where a file image will be saved. The drive for this directory must have enough space to save a complete disk image. One point to take care of is related to compressed NTFS drives. From experience, a compressed NTFS drive has a limit on the maximum file size that may be saved. The maximum is dependant on the size of the CPU memory, and as a rough guide, a 1GB system will have problems with files of about 60GB or greater. The symptom is normally a 'deferred write error'

Log directory

Details of every file recovered is logged. The logs can become fairly large, so adequate space should be allowed. However, the logs will compress well, so it is a good directory to apply NTFS compression to. Under [recovery options](#) it is possible to limit the length of time logs are stored, which again may help conserve disk space.

Scan offsets directory

For some FAT disks it is possible to adjust for areas of the disk that are not in the correct logical location. This is typically due to the hard drive mapping table being corrupted. The offsets are stored in a file in the scan offsets directory

Log Export Directory

Log files may be exported into a CSV format. These files are stored in the directory specified in this section

File Filter Directory

File filter routines are stored as short data files. This is the directory location of where data files are stored.

Search Strings

The search strings directory is where tables of search parameters are stored

Temporary files directory

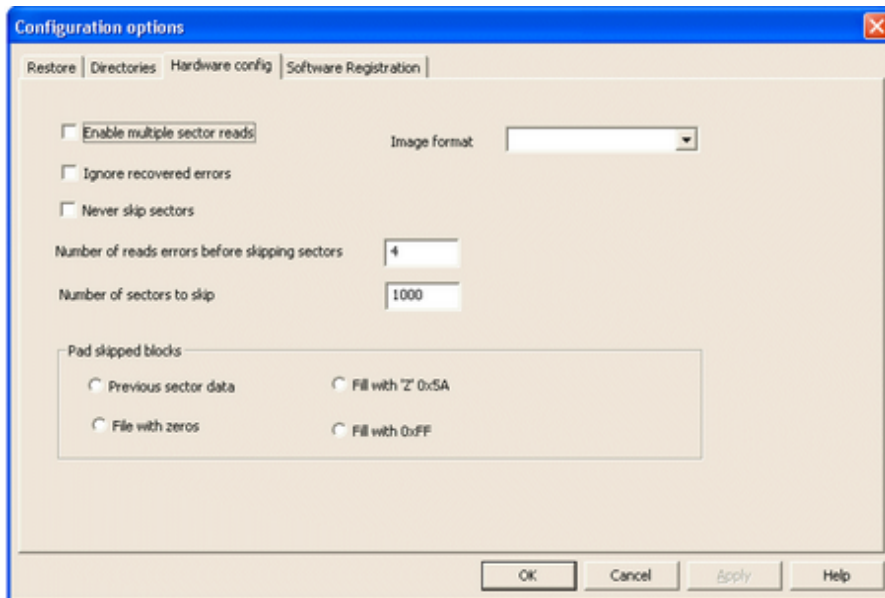
The temporary directory is used for some processes that require temporary files. These files may be deleted, and at times the program will automatically purge the directory. Do not be tempted to store any user data in this directory.

-0-

Hardware config

[Home](#)

The hardware configuration allows users to configure the system to operate as required, in particular when a disk error is detected.



Enable multiple sector reads

This option will place the reading drive into a mode where multiple sectors are read at once. This gives a very large performance benefit over single sector reading. However, when an error is detected, the system will automatically go to single sector reading. It will also automatically go to multiple sector reading after a considerable number of sectors have been read without an error.

Never skip sectors

In this mode, while creating a disk image, the program will never try and skip a sector after a sequence of errors has been found. Forensically this is the secure mode, but realistically, skipping sectors can save a considerable period of time, when creating a disk image, though data may be lost.

-0-

Basic Rules of Data Recovery

[Home](#)

[What to do when media has failed](#)

[Hardware failure - what next?](#)

-0-

What to do when media has failed

[Home](#)

A very important rule of data recovery is never to make the situation worse, and never change anything may prevent the next level of recovery being attempted, if the first level fails. Put very simply, this means that one should never write to a media that has failed or become corrupted. With Read Only memory, such as CDs, and DVDs, this is not a problem. For hard disk drives, this message is extremely important.

If the media is failing, such as a hard disk acquiring more bad sectors, the best first stage is to make an [image](#) of the disk. This image may then be used without any danger of either making the hard disk fail quicker, or fail so that no more data may be recovered. Forensically, this is also a very good move all time spend with the original media has to be logged and monitored very carefully. A true image of the disk, with relevant MD5 hash value reduces some of the chain of custody issues when dealing with a forensic, or legal recovery situation

Data recovery is required when media either fails due to either hardware issues, or software corruption.

Hardware issues

To recover data first one must be able to read the media. A very simple test is to go to the view sector function and try reading a few random sectors. If sectors can be read, then there is good hope of further recovery. If no sector can be read, then one needs to investigate further, but data loss may be the outcome. See [Hardware failure, what next](#) for some ideas

With failure due to hardware issues, the first stage must be to access as much data as possible from the disk, and save on a new storage device, ie another harddisk. The best way to do this is to make a disk image using the [Image and raw](#) recovery function. A curious aspect of hard drives is that most modern ones are always recalibrating themselves. This sometimes shows with a drive that can be extremely slow to read, but after maybe 12 hours suddenly goes quickly. This can also be due

to errors being mainly at the start of the disk. See [Recovery from a drive with many bad sectors](#) for more details

With a drive that cannot be recognised by a PC, but the disk is still spinning, there may be an issue with the controller board. Replacement of the controller is actually very simple, just unscrew about 5 screws. It must then be replaced with an identical card, with identical version. However, do not expect a high success rate for this. If the heads have failed, then a new controller board will not assist. Also, with reference to the recalibration note above, the controller boards may be calibrated so far apart, that no data will be seen.

Opening the drive should never be done. Unless opened in a special clean room environment (ie not an office or domestic room) the drive will be damaged due to dirt in the air.

Software issues

Unlike hardware issues, software corruption is often easier to handle. As always, it is best to make a copy of the drive, so that there can be an unlimited number of attempts to recover data by using different aspects of CnW Recovery software, and sometimes by trial and error with setting parameters. Much of this manual will guide you through different approaches that can be tried.

Corruption, and software failure can be caused by one, or many of the following causes

- Accidental deletion
- Reformating
- Power cuts / surges
- Operating system error
- Removal of media before writing completed, or finalising for CDs/ DVDs
- Reloading operating system
- Boot sector failure

Any many other reasons

Hardware failure - what next?

[Home](#)

If while trying to read a disk with the [View](#) function CnW Recovery software no sector can read, or the sector is just displayed as 5A 5A 5A, it is possible that the media is totally dead. Before giving up, there are several things that can be tried, sometimes media dependant.

The first stage with all types of media is to ensure that the basic reading / interface hardware is working. Thus if trying to read a DVD, test the drive with a known good DVD. If reading a memory chip, try a known working memory chip in the same card reader.

Range of error

It is important to know if the whole media has failed, or just part of it. This can be tested by using the Views sector function, and trying different values, within the range of the disk. Most disks should start reading at location 0 (though some CDs start at 500, and mini DVDs at about 12,000).

The top sector on a disk is dependant on the size of the media. For hard drives, and memory chips, the capacity is normally known, and as rough guide, the top location in MBs is twice the capacity in GBs. For example, a 30 GB disk has about 60 million sectors. A 512MB (0.5GB) memory chip has about 1 million sectors. A sector is 512 bytes.

For CDs and DVDs, a sector size is 2048 bytes (2K). The number of sectors on a CD / DVD does depend on how much has been written but a full CD could have about 350,000 sectors and a full DVD could have just over 2 million sectors. As mentioned above, CDs and DVDs do not always start at sector 0. Sector 16 is normally very important, and so should always be tested

If every sector attempted fails (ie error message or 5A 5A on the screen) then there is not much hope with the present hardware.

Hardware variations

For a CD /DVD it is important to make sure that the drive is a RW drive, and compatible with format being used. Often though, reading marginal CDs can be drive dependant. Thus try the media on a different drive, there may be some luck.

For a hard drive, the preferred way to read the drive is in a USB caddy. Try a different caddy, or possibly connect the drive directly to an IDE connector in the computer

For a USB memory stick check that the connector has not been broken.

Very occasionally it is possible to repair these, but does require a significant element of good luck.

Partial reading

If some sectors can be read, then the next stage must be to make a disk image. Using the Image function an image can be created. If the start of the media cannot be read, it can be skipped. The image will be padded (with 5A 5A). It is also possible to skip a section. If the copy goes very slow due to failed sectors, the copy can be cancelled and started at a higher location. Any missing area will be padded.

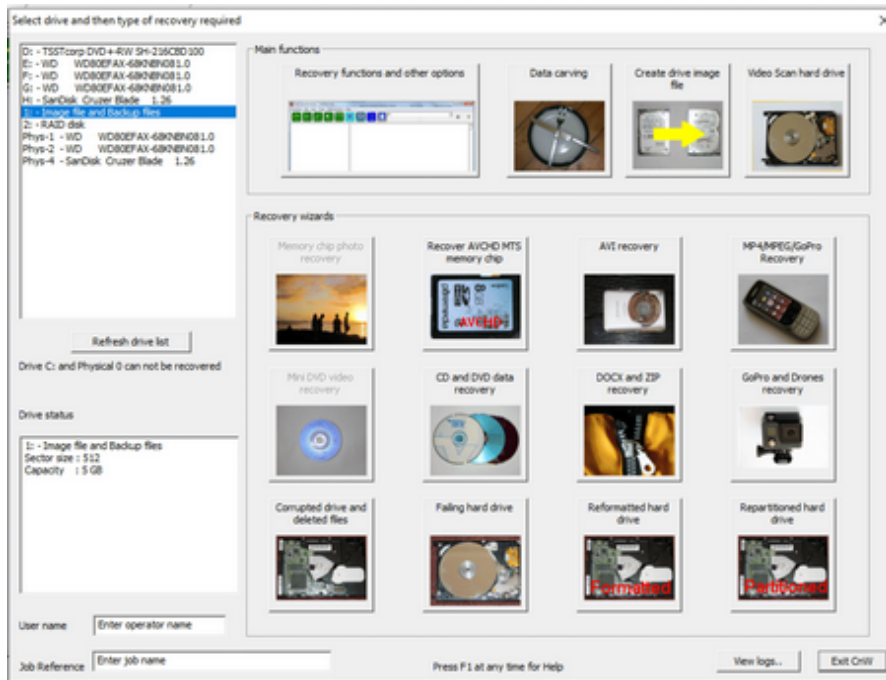
Once an image has been created, a recovery can be attempted, and it is possible that much data can be recovered

-0-

Welcome Screen

Home

The first entry screen for CnW is where the media is selected, and typically the type of recovery. It also allows going straight to the main recovery tools.



To select a physical drive

On program start up, the list box at the top left is filled with all possible drives. If a drive is selected, very brief details are shown below. Also, at this point, the types of functions possible are enabled. The box below gives a very brief description of the drive with partitions and total drive capacity.

It is not possible to select Drive C: or Phys-0 as these are normally the system drive. Recovery should never be attempted on a working system drive as temporary files etc could be written to the disk at any time and potentially overwrite any lost files. To recover such files, the disk must be set up as slave drive on a different PC

User name and job reference

These parameters are used in the log, and as a selection basis for the [forensic report](#)

Main Functions

Recovery functions and other options

The main menu is where all the basic recovery tools can be accessed. There are options to configure recovery functions and modify certain parameters. When a drive has had major failure, or corruption, it is not always possible to determine the exact original configuration exactly, but CnW will allow these relevant values to be entered manually. All the basic recovery functions are selected from this main menu

Data carving

Data carving is the process where a disk of any format may be searched for possible files, based on the file signature. Files are then saved in subdirectories based on file type. This works with all types of media but should be treated as a second option as file names and directory structure is not retained.

Create drive image file

This creates an image file of the media. It is in effect a Unix DD file with a one for sector sector mapping. Blank and unreadable sectors are padded. Incremental imaging may be performed with this function.

Forensic image and scan (under development, due December 2013)

This is a forensics only option. The function will optionally create a disk image file, and then scan for all files, and optionally take hash values of the files, and unallocated space.

Recovery Wizards

Before any routine is run, there is a very simple physical drive test to try and detect if there are significant errors on the drive.

Memory chip photo recovery

For the majority of memory chips, this will be a simple one stage operation. The memory will be analysed, and files recovered. During the recovery process thumbnails of photos will be displayed. This is also an option to help process fragmented files

Failing hard drive

This process will first create an image file of the disk, and then call the relevant recovery routine.

CD/DVD recovery

This function calls the CD recovery function

Mini DVD video

It is very common for mini DVDs to become unreadable. Sometimes due to being taken out of the camera. This function will analyse the disk and produce a complete video file.

Corrupted drive

This wizard calls the main recovery routine with the best settings to read a corrupted drive.

AVCHD memory chip

The recovery scans a memory chip and reconstructs all fragments from [AVCHD](#) files

Repartitioned drive

This is a wizard to be used when a drive has been repartitioned. Attempts will be made to establish any previous partition

Formatted drive

This is a wizard to work on a drive that has been reformatted. The drive is analysed to determine if the drive has had the logical file system changed.

MP4/3GP Recovery

This function does a comprehensive scan of the disk or memory chip and will attempt to reconstruct MP4 video files.

DOCX and Zip recovery - under development

The disk or memory chip will be scanned and all Zip files reconstructed. Fragmented files will be joined and corrupted files will be reconstructed to create a valid ZIP file, but may not be complete

AVI Recovery - under development

AVI files are typical video files from cameras. Being typically FAT32, these suffer from fragmented files and the wizard will help reconstruct such files. AVI is also sometimes used with security camera software.

PC Sleep / hibernate mode

When CnW Recovery is recovering files, the sleep / hibernate mode of a PC is inhibited. This means that long recovery process can be started and they will not be truncated by maybe a 2 hour 'sleep' policy on a PC.

-o-

Unrecognised media

[Home](#)

The CnW Wizard will try and determine what type of disk is being read. If a disk has very few good sectors, or is very badly corrupted, typically at the start of the disk, it can be difficult to determine. It is also unlikely that wizard will manage recovery, so the manual operation will be required.

To determine the type of media does take experience, but the following is a brief guide for the possible types. They fall into two categories, for CDs and DVDS, or for magnetic media such as hard drives and floppies.

CD / DVD operating systems

- ISO9660
- Joliet
- UDF
- Macintosh

For more details on recognition see [How to recognise types of CD/DVD](#)

Magnetic Media

- FAT12
- FAT16
- FAT32
- NTFS
- HFS+ (Macintosh)

For more details on recognition, see [Magnetic media recognition](#)

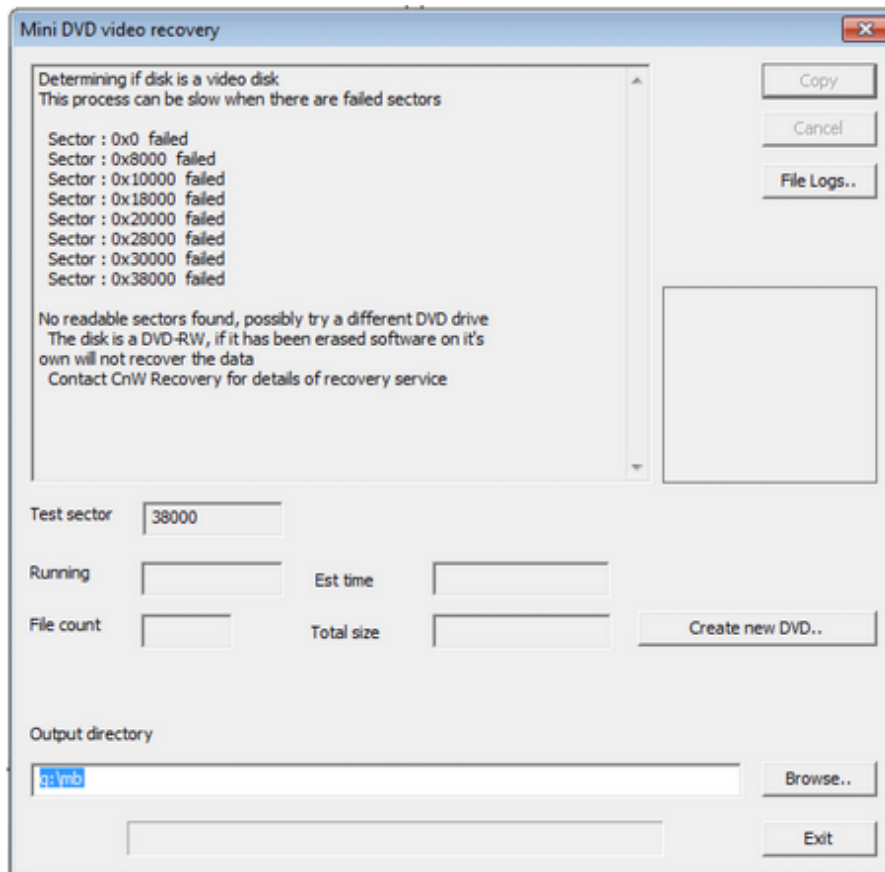
Once the operating system has been determined, it will be possible to start the Recovery function in the correct mode. For CDs, this will mean selecting the correct operating system in the options box. For magnetic media, the operating system will be entered via the partition selection routine.

-o-

Mini DVD recovery

[Home](#)

Mini DVDs store 1.4GB, or about 30 mins of normal quality video. Unfortunately they can fail ofte due to camera or operator error, or just bad luck. This function will analyse the disk and determine if there is viable video on it. If so, it will read and produce a directory with a video disk image.



The process has only only a single prompt, to ensure that the correct out put diectory has been selected.

The first stage is to read areas of the disk to determione if it is a video disk. Many failed disks do not start until about sector 4000, ie approx 0x1000. The next stage is to determine the range of data, testing for both top and bottom locations. If the final location is less than maybe 0x5000 then it is likely that the video disk is not really valid. It indicates that the full disk cannot be accessed by software alone. CnW Recovery do provide a recovery service for such disks, which is performed using specialised hardware.

The Create image file first option will generate and image file on the output directory. This is a useful backup of the DVD disk.

Once the output directory has been selected, the program reads the disk and

extracts all MPEG files. These are stored in a directory within the output path. The final directory structure is as below

x:\chosen_dir\!video\mpeg\VIDEO_TS

The !video directory will have a MPEG and IFO directory to store the raw MPEG and IFO files

The VIDEO_TS directory will have a recreated disk image, and recreated IFO files - the original IFO files are ignored as often the data is not actually complete.

Once a recovery is complete, there is an option to [create a new video DVD](#). This DVD will be playable in a standard video player.

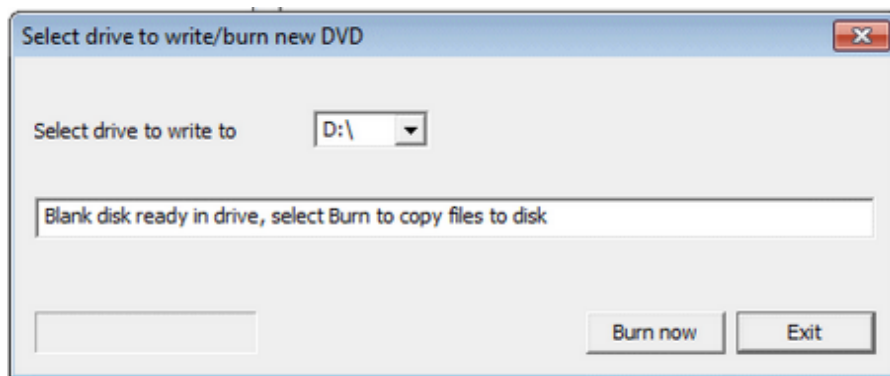
-0-

Create new video DVD

[Home](#)

Once the wizard has recovered video data, the data is stored in a file structure suitable for burning to a new DVD. This function will write the DVD as a simple one click procedure.

The program will search for suitable drives, and give a list of such drives. Once the relevant drive letter has been selected, it will wait until a blank DVD has been detected. At this point the Burn now function can be selected, and a new DVD created that will be compatible with standard video recorders.



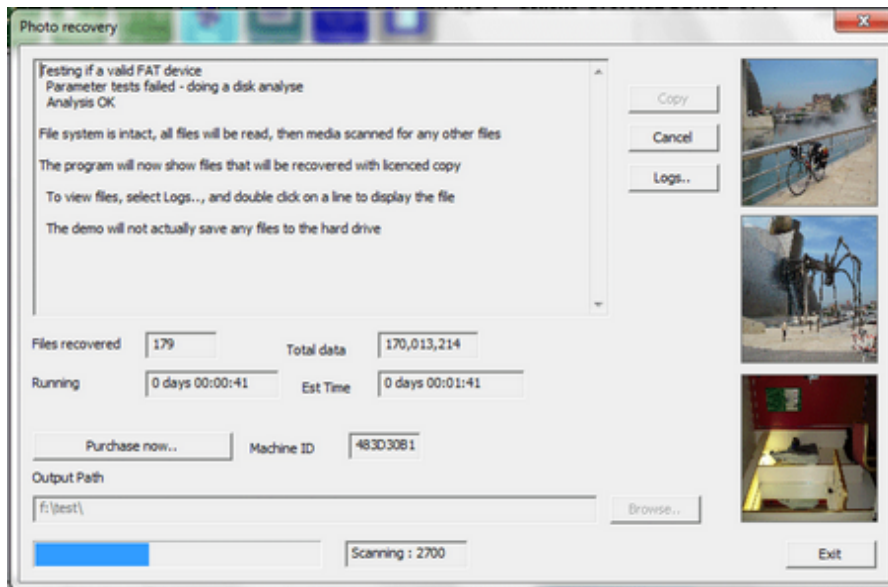
-O-

Photo Recovery

[Home](#)

Memory chip and photo recovery is a routine aimed at camera memory chips. It will be selected for any device upto 64GB which is either as FAT device, or has no recognisable file system (many camera memory chips have had the start of the chip overwritten).

The routine is designed to be as automatic as possible, and the chip will be analysed before the type of recovery chosen will be attempted.



During the recovery process, example recovered photos will be displayed on the right hand side of the screen

The option to process fragmented files is applied after a recovery scan. It will detect which files are not valid, possibly due to fragmentation, and attempt to create new files from fragments found on the disk. The success rate does depend on the fragments being present in the first place, and also on how fragmented the files are.

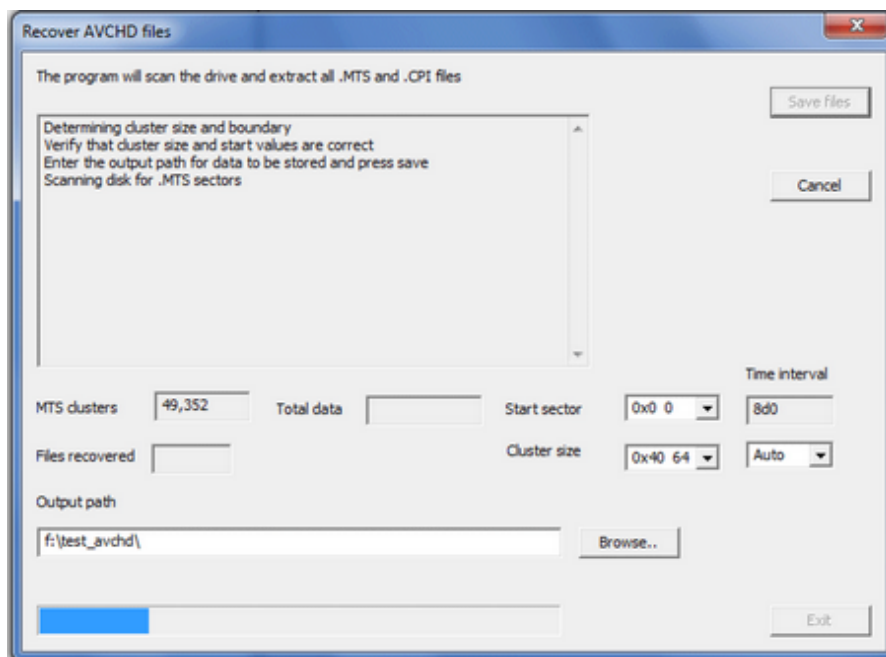
-0-

AVCHD recovery

[Home](#)

Video camera often use FAT to record data. When deleted, all details of any data fragments is lost. Also, on some cameras, all information about the location data is stored in is also lost. CnW have therefore created a special, dedicated function to scan the memory chip and recover all AVCHD video clips. Unlike many other recovery programs, it will produce a number of long clips rather than 100s of short clips that have to be joined together by hand. The process allows for fragmented and deleted files.

The routine was designed for memory chips but with V3.91 the 128GB limit has been overcome. Recovery from hard drives is not recommended, but only due to speed issues. The routine does examine every sector and for a 1TB drive this will take a few hours. However, good results will be achieved.



The program is a two stage process

- The chip is scanned for all MTS clusters
- The clusters are then sequenced into long data runs
 - CPI files are recovered
 - MPL files are recovered

Where CnW Recovery scores over other recovery programs is that it examines the whole memory chip first. It logs the start and end of each cluster of video. From this table it can build complete files, rather than just short fragments. On one real life example 8GB memory chip, there were about

500 fragments, which competing software produced as 500 MPL files. CnW reconstructed the correct number of about 60 files. Where possible (ie the information exists) the original dates of the video files are extracted.

Setup parameters

There are two important parameters that must be set correctly before starting. The first is the sector start and cluster size. For FAT chips these are often (but always) determined by the program. The start sector is an offset of the cluster on the disk, and is modulus the cluster size. If the values are wrong then there will be many more files produced. For most memory chips, the values appear to be start sector 0, and cluster size of 0x40 64.

For non FAT disks the cluster size and start sector may be different. eg NTFS may have a typical cluster size of 0x8 and often a start sector of 0x0 or sometimes on XP drives 0x7

The time interval is the interval between frames of video. Auto may be selected, or fixed values of 0x8D0 or 0x69C, 0x5ab, 0xb96 may be selected. If the value is wrong, the symptom is many short files being found rather than 10s or 100s or reasonable sized files. If none of the options work, please send some sample (short) .MTS files to CnW and a new value will be added.

How to view AVCHD files

AVCHD is a complex format that is a mix of MPL and CPI files. All the video is stored in MPL files, and these can be viewed through programs such as MediaPlayer - but only on Windows 7 (ie not XP). CnW recovers files, but does not create valid file names, ie the MPL and CPI do not necessarily tie up. One suggested method of recovery is to copy the recovered, complete files back onto a memory chip, and get the video camera to re-index the files. This should produce a memory chip in the format as it was before the files were deleted, or corrupted.

To view a files on an XP PC you will need to download a compatible viewer - do a Google search for possible viewers.

NB Never write to the original problem disk/memory chip unless you have a copy of the disk. If it is required to write to a memory chip to re-index in a camera, either use new chip, or make sure you have a complete, secure image of the original chip (using Image Disk function).

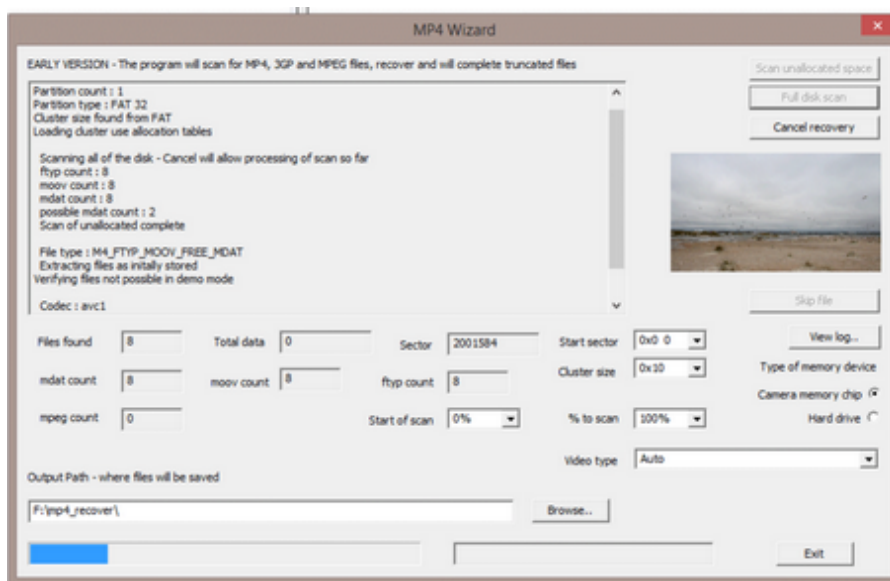
3GP, MP4 and MPEG recovery wizard

[Home](#)

The wizard is largely aimed at recovering video files from corrupted or deleted memory chips that were formatted with FAT32. The process involves scanning the complete device so will slow if a large drive is analysed this way. The wizard works for most files with typical extensions of .MP4, .3GP, .MOV (although .MOV covers several different formats).

3GP and MP4 files are often not stored sequentially on a disk and so special tools have to be developed to recover the files. This wizard is designed to detect, analyse and recover as many files as possible. This can include reconstructing missing elements of the video file, in particular the moov atom which is essential to viewing the video.

The wizard is primarily designed for memory chips, but will also work on hard disk drives. Files on hard disk drives will typically be sequential, while deleted files on a FAT32 chip are typically not sequential. The wizard aims to detect the file type and create a playable video. For both media types, attempts will be made to reconstruct videos that are not complete.



The main way to operate is via a Raw Scan -which scans the complete disk as described below. An option under development is for Scan and Save files which will concentrate on just repairing damaged files, before doing a scan of the unallocated area of the disk.

The wizard works in a few stages

Stage one:

The complete disk is scanned for elements of video files. Typically it will be looking for MP4 tags for the main atoms, such as 'ftyp' 'mdat' 'moov' and

'free'. each location is then saved in an internal table

Stage two:

A moov element is looked for and if a sequential complete one is found, details are parsed. This indicates the type of camera, type of file, and sometimes the structure expected on the disk

Stage three:

At this point, possible files are constructed starting with a 'ftyp' atom, and then looking (typically) sequentially for the 'mdat' and 'moov' atoms

Stage four:

The file is parsed to see if valid. If it is not valid then full recovery takes place. At first attempts are made to tie the mdat and moov atoms together. If this fails, attempts will be made to reconstruct a moov atom from scratch. This last stage does require a valid moov atom to be present on the disk.

Thumbnail

When the wizard makes a good recovery a thumbnail image will be displayed - even on the demo. For the demo it gives a high level of confidence that the video will be fully recovered. (Not available on Windows XP systems)

Warning messages

In the scanning process certain inconsistencies are sometimes detected, and a ***WARNING*** message displayed

****WARNING *** A free atom found larger than cluster size
It is possible the cluster size is incorrect - Press F1 for more help
Try cluster size of 0x40*

This error indicates that the cluster size is not consistent. A FREE atom is normally smaller than a cluster, and used to pad to the end of a cluster. If the video has been moved from its original memory chip, then there may be a different cluster size. This would also be true if a memory chip was copied to an NTFS hard drive. In this case, the warning message can be ignored. If on the original memory chip, the 'Try cluster size of 0x??' is a guide value to be tried.

****WARNING *** ftyp atoms have been found not on cluster boundary
It is possible the cluster size, or start sector is incorrect. Press F1 for more details
Try start sector of 0x4
And/Or try smaller cluster size*

This error indicates a possible problem with the start sector value. The ftyp atom is normally only found at the start of a cluster. Any other location indicates a possible problem. As in the previous warning, copying files to a hard drive can cause false positives.

Some of the file/camera types currently supported. Many other cameras will match these variations

Camera

Canon EOS700
Canon SX600 HS
Canon HF-G30
GoPro Hero - Black and Silver, all variations including high and low resolution files
Samsung HMX-H300
Kodak Zx1 Pocket Video Camera
Panasonic DHC
Sony PWM-F3

Video format (codec)

AVC1
MP4V
JPEG - under development

Logical file layouts supported

The following [link](#) is one that will grow on a regular basis until all common combinations are fully supported. The level of recovery will vary on each type but will eventually include repairing the following types of failure

- Finding each cluster from a fragmented file (often deleted on FAT32 disks)
- Creating a moov atom when it is missing
- Creating a ftyp atom when missing - this in effect will allow raw mdat data to be displayed

The aim for each disk type is to make the video playable even when areas are missing.

MPEG recovery

This function also looks for and combines MPEG fragments. It is intended to be used on hard drives and will attempt to find runs of MPEGs and then join them in to sequences. The matching is not always perfect, so sequences should be checked for false matches.

Cluster size issues

Sometimes warning message will displayed indicating that the cluster size, or start sector may be incorrect. The routines expect a FTYPE atom to be at the start of a cluster, ie byte offset 4. It also expects that if there is a FREE atom it normally only goes to the end of the current cluster. If these conditions are not met, a warning message is displayed.

The default cluster size and start sector are generated from the file system information, but is not always correct. This can be very true if images have been moved to a different type of drive.

The solution is not always obvious, and may be a combination of cluster size and start block. However, for a modern memory chip, the cluster start is normally 0x0 and the cluster size for a FAT32 is very often 0x40. It may be necessary to try a few variations

Repair not implemented

At times a message such as "Repair not implemented M4_FTYPE_FREE_MOOV_FREE_MDAT" will be displayed. This means that this possible repair or reconstruction mode has not been implemented yet. Contact CnW if it is required. The intention is to cover all permutations, but development time is not unlimited.

File type not recognised

At times the program may not recognised the structure of the data and will request to send a diagnostic log file to CnW. The file is mp4_scan_<date>.\$\$\$ eg mp4_scan_20130528.\$\$\$\$. This stores the locations of each ftyp, mdat etc found on the disk scan and will help CnW analyse the memory structure. There is no user data in the file, just sizes and locations. NB, because the file is stored in the CnW Temp directory, it is cleared down each time the program starts.

Scan range

When processing camera memory chips generally speaking it is normal to scan the complete chip. Hence the options for start of scan, and % to scan will 0 and 100. If dealing with a large hard drive, the scan may be very slow. For a 3TB drive we are looking at many hours just for the base scan. For this reason, it is possible to select just an area of the disk to scan, maybe the first 5% as in the example in the screen image above.

Type of memory device

There are two types of file system that can be recovered from, the original camera memory, and a logical copy, typically a hard drive. Each will expect data stored in a different way. For camera memory, it is often for

the basic sections to be stored out of sequence, and at times the moov atom may not have been generated. For a logical drive, the file is normally in sequence, but may be fragmented as the result of copying files to a previously used disk drive. Both of these approaches require slightly different recovery processes, by setting the option to either camera memory chip, or hard drive the best results may be obtained.

- Camera memory chip - when processing the chip that was in the camera, ie will be physically out of sequence
- Hard drive - when processing files that have copied to another device - ie will be physically sequential

Video type

The default setting is Auto. For memory chips this is probably the best setting. For hard drives there is a significant chance that the drive will contain more than one type of video. The option of video type will allow focusing on the relevant type of video file. The problem (for the user) is to establish which type of file is required. The only safe way is to look at a known good file and hence determine the correct type. For more details look at [M4 disk layouts](#) or the list below. For hard drive, this option allows the selection of specific file types. Thus if a Canon camera is to be recovered, the correct file type can be selected.

Save 2GB Image

This option will save the first 2GB of the memory chip image, to the selected output directory. This works, even on the demo version. The purpose for this option is to help solve any problems. The 2GB file can be sent to CnW Recovery (possibly using the free program www.wetransfer.com). The file name will be called 2g_image_1234abcd.img where 1234abcd is the Machine ID code.

Development Status

The Wizard function one of many elements. Only certain stages have been completed and the table below shows the status for each type of video. There are three basic processes, and they are tried in turn. If a video can be verified, nothing more is done, if it fails verification, then the next stage is processed

- Stage 1 - Recover. Assume that the data is in the normal sequence for the format
- Stage 2 - Repair. Assume that the basic sequence is correct, but the moov or mdat has become fragmented. ie find all the pieces and put them back
- Stage 3 - reconstruct. This is when typically the MOOV atom is missing (or too badly corrupted). Data that looks like video data is parsed, and the MOOV atom is reconstructed. For this to work, there must be at least one valid video on the memory device

File type	Recover	Repair	Reconstruct
*M4_FTyp_MOOV_MDAT	Yes	Yes	
*M4_MDAT_MOOV	Yes	Yes	
M4_DATA_FTyp_FREE_MOOV_MDAT	Yes	Yes	
M4_DATA_FREE_FTyp_FREE_MDAT_MOOV_FREE	Yes	Yes	Yes
M4_FTyp_MDAT_MOOV_FREE	Yes		Yes
M4_MDAT_FTyp_MOOV_FREE	Yes	Yes	
*M4_FTyp_MDAT_FREE_MOOV_FREE	Yes	Yes	
*M4_FTyp_MOOV_FREE_MDAT	Yes	Yes	
*M4_FTyp_FREE_MOOV_FREE_MDAT	Yes		
*M4_FTyp_MDAT_MOV			Yes

* represents a format typically found on hard drive, rather than on raw memory chip

The above list does change on a regular basis. For some of the repair and reconstruction it is dependent on the type of video codec used

File type and possible camera

Camera memory chip

M4_MDAT_FTyp_MOOV_FREE

Canon EOS /Rebel range, eg 600D, 70D

M4_FTyp_MOOV_FREE_MDAT

GoPro Hero 3 Black

Ambarella

M4_FTyp_MDAT_MOOV

GoPro Hero 4 Silver

Nikon D5100

Panasonic DMC-GH4

Canon SX100 HS

Hard drive formats

M4_FTyp_MOOV_FREE_MDAT
Canon 700

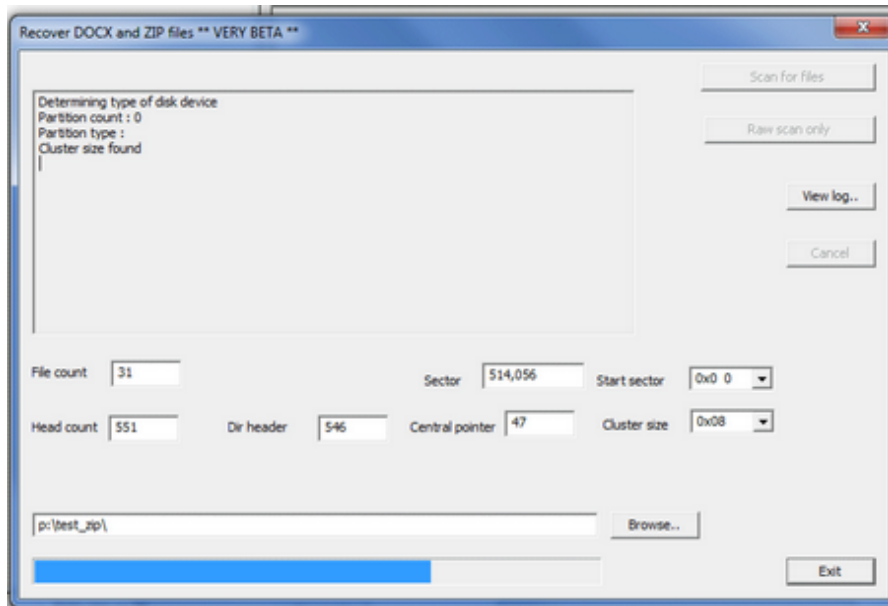
M4_FTyp_FREE_MOOV_FREE_MDAT
Kodak Zx1 Pocket Video Camera

-0-

ZIP and DOCX recovery wizard

[Home](#)

The wizard is designed mainly for use with memory chips and other FAT memory devices. The process including scanning the complete disk so could be slow if a large hard drive is analysed this way.



The important parametrs to set are the start sector and cluster size. When possible these are automatiucally determined.

On scanning, the program will scan the complete disk - thus for a large hard disk it may be slow which is why this function is most appropriate for FAT disks that tend to be memory chips or smaller.

On scanning three elements will be searched for

- Head count - the is a PK 0x03 0x04 header found at the start of each block of compressed data, and contains the file name.
- The dir header is the PK 0x01 0x02 header. This is the directory entry for each file and gives the size and offset within the main ZIP file
- Central pointer. This is at the end of the Zip file and is a PK 0x05 0x06 header. It points to the start of the directory.

After scanning the program will attempt to recreate Zip files. This is process where a starting PK 0x03 0x04 is read and each file within it is read. By knowing the length of the file it is possible to determine the location of the the next header. Having scanned every cluster it is possible to find a cluster with a header in the correct location. The chance of a false positive is not very likely, but to trap these the CRC value is tested, and if it fails, another cluster

is tested.

-0-

Video scan of hard drive

[Home](#)

UNDER DEVELOPMENT!

This wizard is designed to recover specifically video files from a hard disk drive. CnW has several wizards for video recovery, but these are designed with camera memory chips in mind. The videos on these chip are often out of sequence, but at the same time, camera memory chips are of limited capacity, eg 64GB. Hard disk drives are commonly 2-3TB, and growing every year. Also, files on the hard drive tend to start as sequential files, but can still be fragmented or damaged.

For camera memory chips use the MP4 wizard

The wizard is used to scan the hard drive for specific types of video file, save, reconstruct and possibly repair. Hard drives tend to save files in sequence, and so recovery is very different to memory chip[recovery

Current status - September 2020

The drive will scan and search for video files in certain formats

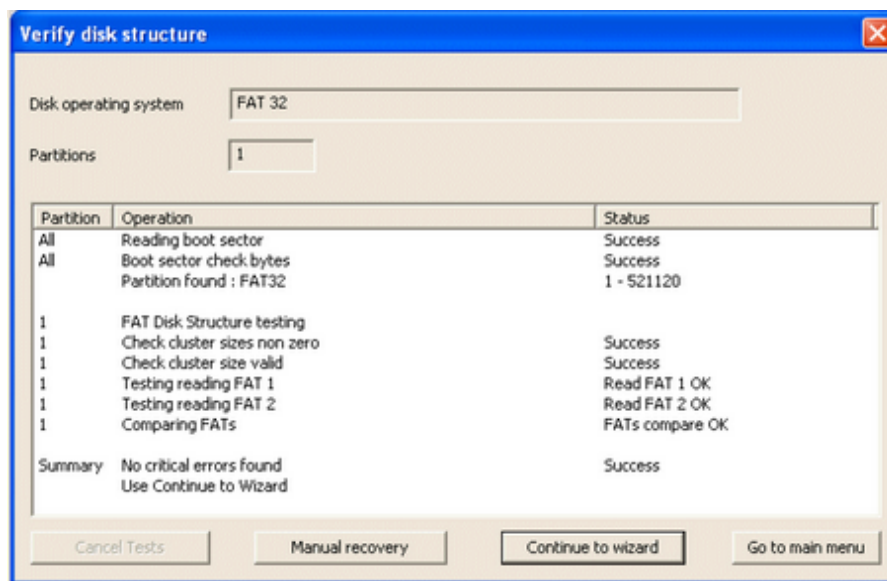
The first version that has been implemented are files that are saved as FTYP-MOOV-MDAT, and ones that have video frame lengths saved at the start of each video frame. This could be a Canon style of video. The program will detect the file start (a FTYP atom) followed by the MOOV. It will process the MOOV to find the start of each frame and then search for video frames in the following data. It assumes that the file is in sequence, although it may be fragmented. ie it only searches forward for the next possible video frame.

-0-

Verify disk structure

[Home](#)

The verify disk structure is a series of tests to see if the most important parts of the disk are correct and valid. If the tests are successful, then the Wizard will probably recover the required files. If the tests fail, then the program will recommend a more hands on approach to recovery using the manual tools. The type of tests carried out are based on what an experienced data recovery person would do to determine the status of a disk before attempting and further recovery.



The tests are dependant on the disk operating system, and will automatically sequence through for all detected operating systems. Thus some disks have a mixture of FAT and NTFS.

Each test will be described briefly, and a status given.

There are a different series of test for each type of media. For CDs, various areas of the disk are examined and for a blank CD-RW, it will suggest that an [unerase](#) process could be tried.

The tests are done for all partitions, and so the partition being tested is shown in the first column. Details of each test are described below.

Boot sector tests

For any disk to read logically it must have a valid boot sector. This is therefore the first test of any data recovery routine. The sector must be readable, and contain valid data. For instance, the check bytes test (see in

the picture above) ensures that the final two bytes of the boot sector do contain the bytes 0x55 and 0xAA. It will then decode the 1 to 4 partition tables stored at location 0x1be in the boot sector.

If the boot sector has failed, then the program will indicate that the partitions function should be called to reconstruct a partition table.

CD and DVD tests

Many CDs and DVDs fail due to not being to read the start of the disk. The tests performed are to determine how much of the start of the CD/DVD can be read physically. For video mini-DVDs, it is common for the start to be blank, and then video to be found at about sector 0x4000.

FAT Tests

A FAT disk is read using information such as cluster size, and FAT tables. This information is read from the control sector at the start of the FAT partition. Simple tests are carried out here to make sure that the parameters are within sensible ranges. For instance, a cluster size must always be a multiple of sectors, such as 1, 2, 4, 8. If a cluster size of 3 or 17 is found, this is invalid. The disk analysis routine in the FAT handler will assist in resolving this problem, but the straight forward wizard will not work.

It is also important that the FAT sectors can be read. The test will attempt to read both blocks of FAT sectors. Each block should be identical, so it also compares them. Any errors detected will indicate problems with the disk. These may be overcome by reading the disk, ignoring the FAT, or reading the disk and using FAT2. When reading the disk, the FAT handler, when it finds a failed sector in FAT 1 will automatically look at FAT 2 to see if the matching sector can be read.

NTFS tests

The most important part of a NTFS disk is the \$MFT file. This stores all file entry details. The verify routine does a simple physical read of this file, and a verification of the first MFT entry.

At the end of the tests there will be a suggestion as to the next stage. Typically it will be Continue to Wizard. Other options can be Manual Recovery, or Go to main menu.

Manual Recovery will exit the wizard and select the Recover Function. At this stage one will have an indication of possible problems with the disk, as well as areas that are considered to be valid

Go to main menu will exit the wizard and go to the main program menu - no further assistant prompts will be given.

-0-

Physical Media Test

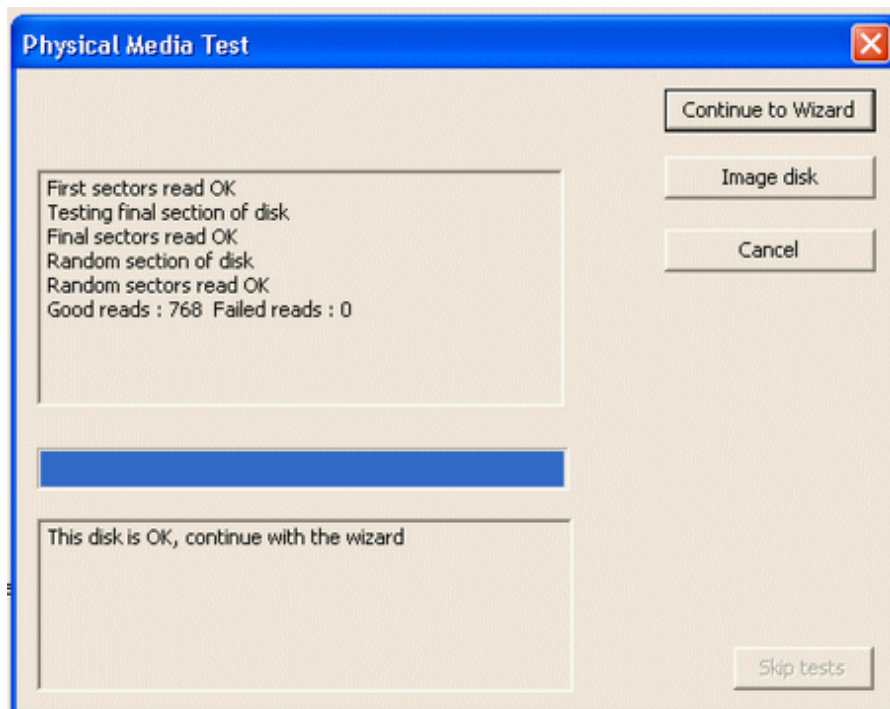
[Home](#)

When doing a data recovery on a hard drive it is extremely valuable to perform some basic tests on the drive to determine the best way to progress with the recovery. The Physical Media tests do this by reading different areas of the disk to try and detect if there are likely to be a significant number of failed sectors. The tests are reading from the start of the disk, the end of the disk, and then random areas of the disk.

On a good disk, the result will be shown as below which indicates that no errors were detected, and reading was fast. On this type of disk, the program will automatically continue to the Wizard

On a problem disk, there will be indications of failed sectors, or sectors that were slow to read. There should never be indications of slow reading on first and final sectors, but occasionally disks will indicate a slowness on random sectors. This is acceptable. Each test will time out after a minute, and if it does time out, this indicates that there is a physical issue with the disk.

Depending on the results, a choice can be made to image the disk first, or try a straight recovery of the data.



Continue to Wizard

This is the standard option if the physical test looks OK. If there are no more than one or two errors, and the reading is reasonably fast, then the proceeding to the wizard is the best approach. If there are too many errors

detected, this option will be disabled.

On may formats, the next stage will be a quick evaluation of the disk structure.

Image Disk

If the disk is very slow to read, or has many errors the the best procedure to follow while recovering a disk is to create a disk image. By clicking on this button, the program will go directly to the [disk image](#) function.

Cancel

This will return to the [Wizard](#) entry screen

Fake memory chips

There have been several cases where data was apparently lost on a memory chip but the problem is that the memory chip is a fake chip. ie It is marked as maybe 32GB, and when formatted looks like 32GB. However, internally it only contains 4GB of memory. When written to, all looks OK until more than 4GB is written. On ones seen, the data is then wrapped around maybe just 16MB or 64MB of data area. No errors are seen on writing, but when reading back, in the example above, upto 28GB of data will be lost.

The physical media test will try and detect such chips. If found, it is just bad news. Recovery can be attempted, but expect to loose all data after the start of the physical memory.

-0-

Failing disk drive

[Home](#)

It is common for a disk drive to be failing - ie it has a considerable number of bad sectors. A very high level of recovery can often be achieved, but it is very important that the disk is not stressed any more than necessary. Experience will also show that typically the area of disk to fail first is the directory area, which is also the most useful area of the disk. In normal disk reading, it is common to read elements of the directory multiple times, partly to reconstruct directory structure. The safest way to progress with such a file is to create a file image first, and the intention is to read every sector only once

-0-

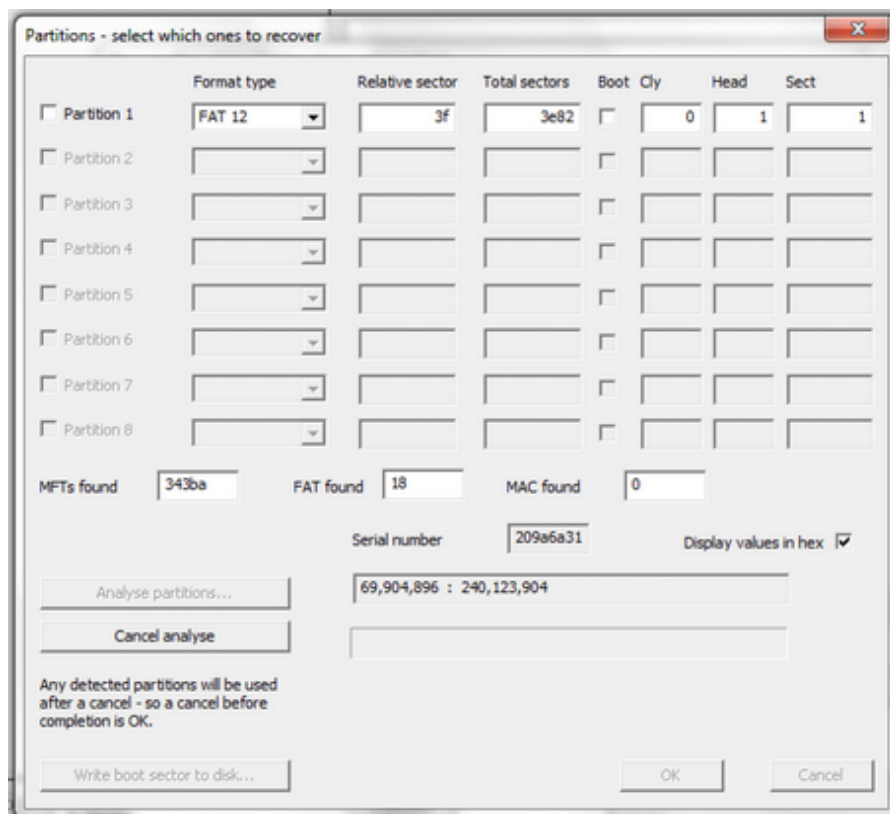
Formatted disk recovery

[Home](#)

It is common for a disk to be formatted, but as long as it is not a full format (erasing all sectors), most data can still be recovered. This wizard function works through the stages required for recovery.

The stages of analysis and recovery are as follows

- Determine if current file system is as expected, if the drive is currently FAT, was the original disk FAT
- If it is unknown, or a different file system is expected an analysis of the drive is carried out
 - The analysis will scan the drive for old partitions, but in particular will look for MFTs, old FAT directories and old MAC catalog entries
 - The end result will be the partition screen shown below and it is clear that a current FAT has in the past been a NTFS drive



- The next stage is to recover data in the format determined from the above.
- As the disk is no longer the selected logical format, it will be necessary to do more analysis on the drive to determine the correct parameters.
- There is an option to Skip program files. This will skip files such as

.dll, .exe, .cab, .ini, .sys. Generally these types of files do not contain user data and are not required.

Program features

The program is designed to assist with formatted disks, and ones that the operating system has been reloaded. In these cases the directory structure is often lost and recovery results in many 'lost_dir' directories. Also many files found are not actually required so a filter is set to remove many files that are not normally required on this type of recovery. The list includes

.dll, .exe, .msi, .cur, .fon, .xml and others

It also skips files found in typical system and program directories, such as

windows, program files, program files (x86)

The intention of the above filtering is to try and just find useful file

As an extra feature for Forensic users, it is possible to select (at the stage where the file system options are shown) an extra filter function of removing system files based on their MD5 hash value. Hash tables can be downloaded [from NSRL](#) and used to skip all known system files

-0-

Partitioned disk recovery

[Home](#)

This function is used when a disk has been repartitioned, possibly with a different operating system. It can be possible to scan the disk to detect a file structure from the previous partitioning. An example may be if a FAT32 has been reformatted as an NTFS disk.

The basic tool for recovery is the Partition scan function. This is a function that scans each sector of the drive and determines if it is a possible start of a partition. This logic is different for each type of partition. #

NTFS partitions

There are two main ways to detect the start of a NTFS partition. This can be to find the [partition boot sector](#), or to find the start of the [\\$MFT](#) file. One problem with both of these approaches is that many false positives can be found. Thus verification is required.

For a BPB, it is important that the cluster address in offset 0x30 must point to a \$MFT sector. If the BPB does not point to a \$MFT file, then this potential partition boot sector is ignored.

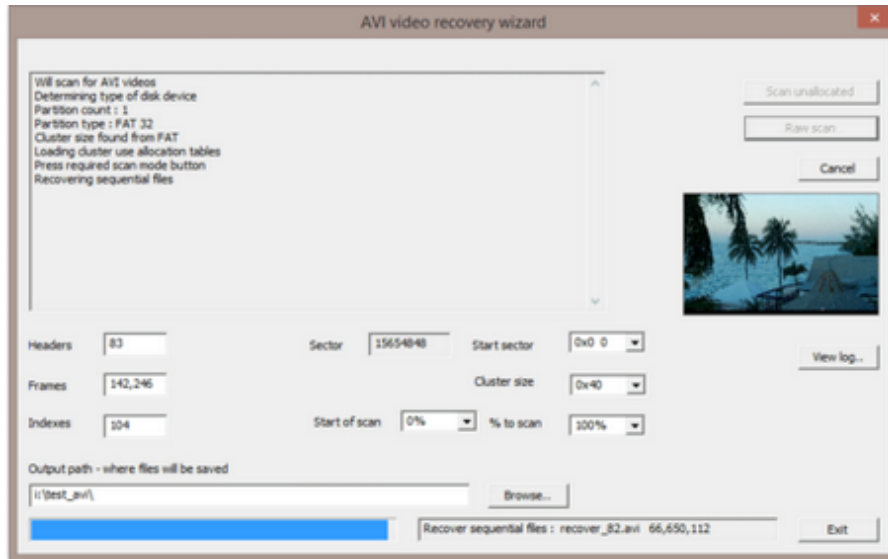
The second approach is to search for all MFT entries. These are then parsed to detect the start of an \$MFT. An \$MFT entry has a pointer to itself, and from there the location of the partition start can be determined. Knowing this value, the 5th entry in the \$MFT file can be read and the location of the '.' root directory index file found. This location is tested to see if it does start with INDX. If it is correct, then the possible partition start sector can be set. If INDX is not found, the \$MFT is treated as a false positive, and hence ignored.

-0-

AVI Recovery

[Home](#)

AVI Wizard will recover files that have been deleted from memory devices. It will also handle video files that have become fragmented.



The program starts by scanning the complete memory chip and then reconstructs video files in the following sequence of events

- Finds file headers and assumes a sequential file
- If the sequential file fails, it will search for the index, and then search for each video cluster
- If the sequential file fails, and no index can be found, an index will be generated from the video data that exists
- If just indexes are found, the file will be reconstructed from the index

There are two types of index for AVI files - CnW support both types, but with a file size limit of 4GB - this will be supported later

-0-

Forensic Data Recovery

[Home](#)

What is the difference between Forensic Data Recovery (FDR), and normal Data Recovery?

There are many answers to this question, so the following summary is just one solution. Both tasks are required to recover files when they have been lost, corrupted, deleted, or just cannot be read by the operating system. The desired result is a selection of files that can then be read.

Both solutions can use the same technique of tolerant reading, searching for lost directory entries, or just a raw file search using signatures - often referred to as data carving. The difference comes with the associated documentation and monitoring of how a file was recovered. This can often mean logging the sectors that made up the file, and also retaining the metadata from any directory entry. For a secure recovery, it is advisable to create a MD5 (or SHA-256) hash value of the file data in order to trap any subsequent, accidental or deliberate changes to the file. CnW Recovery will always log an MD5 and a SHA-256 hash value for all files recovered

Forensic investigation of a file is not part of FDR. Thus one is not interested in how a file has been edited, but all dates relating to how and when a file has been written to the disk drive are very important.

CnW Software has a comprehensive range of logs which track all sector numbers, fragments, dates etc. There is also a [report generating](#) function to produce an XML report.

When do an FDR, it is essential that the data on the drive is not changed. Thus a Write Blocker should always be used, and identical copies of the disk should be used, after the original imaging.

-0-

Recovery Functions

-0-

Getting started - General data recovery

[Home](#)

Learning to use any software always takes time. Some applications are very straight forward because their function is known. For instance to write a simple letter in a word processor should be simple straight out of the box. For data recovery, the situation is more complex. This is because each piece of media can fail in different ways. No single approach will always work. The following guide is intended to assist users find their way around the program, and bit by bit see what options are available and should be used.

To use this as a tutorial, follow each stage by clicking on the links. Each stage there are several options, and these are discussed to assist.

There is also a chapter with extra, [different tutorials](#).

The section [Recognising sectors](#) will be of use for any user not very familiar with the elements that make up a disk

Stages of data recovery

1. Connect suitable drive to computer with CnW Recovery software loaded. [Installation](#) and [Media detection](#) links may be helpful
2. Ensure that the drive is being viewed - use [View sector](#) to look at a few sectors
3. Run [Wizard](#) and select drive to recover data from

If wizard does not work, use manual mode

1. Select Recover files icon
2. For a hard drive, it is worth checking the [partitions](#) are valid
3. One of the following Recovery options will be displayed
 - [NTFS recovery](#)
 - [FAT Recovery hard drives and memory chips](#)
 - [CD Recovery](#)

For disks that cannot be read logically

Certain disks may have been corrupted, or damaged to such an extent,

that logical reading is not possible. For these disks, the next solution is the [Data carving](#) option.

For disks with many physical errors

It is common for disks to fail partially. In this mode, many sectors will read, or read after several retries. Some sectors will be unreadable. Trying to recover from these disks can be very slow, in particular if there are failed sectors within the directory area. CnW has several tools to assist in creating a workable disk image.

To recover deleted files

Accidental deletion of files is a very common problem. Recovery is normally possible, as long as other files have not been written to the disk. For CD-R, recovery should always be possible, but for CD-RW it is dependant on what has been written to the disk after deletion. The first stage for recovery can normally be the Wizard, and select the Recover Deleted File option.

To recover formatted disks

If a disk has had a complete format (and not a quick format) then your data is lost. There are stories that using special equipment, with the budget of the CIA, that it is possible to detect residual values of a previously written byte. The argument, is that if a bit is changed from a 0 to a 1, then it may only go to 95% of the expected value. With the density of modern disks, and the recording methods used, I would suggest that it is not actually viable to attempt to recover more than maybe one or two sectors. A typical JPEG photograph is about 1MB, or 8,000,000 bits, which is probably about 12,000,000 flux transitions on a disk. That is a lot of work. For more details, look at www.nber.org/sys-admin/overwritten-data-guttman.html

If the disk has had a quick format, then normally only the directories and sector use tables are initialised. The old data often still exists and can be recovered. The best approach to recover data varies slightly for each operating system, but for NTFS, searching for all MFTs is a good start. On all formats, there is the option to recover unallocated space, but this does have the problem that typically, filenames will be lost.

To recover files from unallocated space

Unallocated space is the area on a disk that the operating system says is free, ie there are no files in it. On a brand new disk, this space will normally be blank, or just the values used by the disk initialisation program. On a used disk, this space will often contain files that have been deleted, or possibly moved with a defrag program. The recovery of files from this space does depend on the operating system used, hence the normal recovery mode should be selected, and then the option to

recover files from unallocated space should be enabled. All files from the unallocated space will be stored in a main directory of !recover. As there is no information on file structure, the recovery is basically in [Data carve](#). Another way to recover files from the unallocated space is to scan the disk for directory stubs, or MFTs (depending on operating system). This will recover files in a more logical way, but the option to scan the unallocated space will still apply after all other files have been recovered.

Raid Disks

Raid disks can be recovered once an image has been created - see [Disk image](#) for more details. For more comprehensive raid recovery, see the [raid option](#), a chargeable option

-0-

Typical data recovery procedures

[Home](#)

Although each type of data recovery may seem unique, fortunately there are patterns. This page is an index to sections that describe each pattern in detail. They fall into groups for each type of media, and each type of logical format

CD / DVD

[Camcorder disks](#)

FAT 12/16 and 32

[How to recover fat disk when boot sector and one FAT is missing](#)

Missing directories and files on a FAT disk

[How to recover a FAT disk when boot sector is missing](#)

[Deleted FAT32 file recovery](#)

NTFS

[Files lost when NTFS reloaded](#)

General

[How to recover corrupted partitions](#)

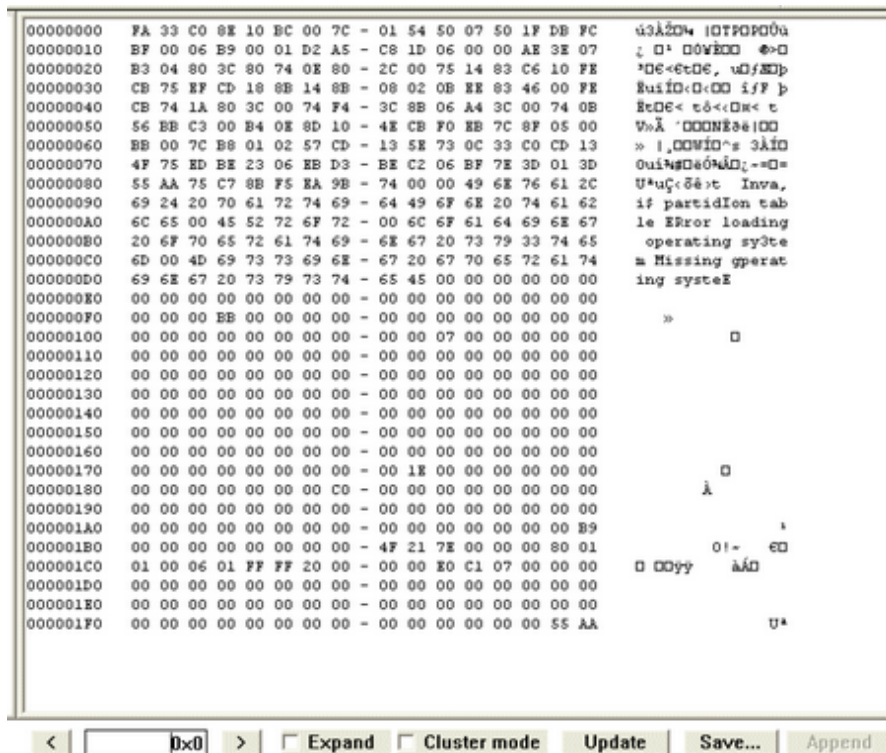
[How to recover when a partition has been deleted](#)

-0-

View sector on hard drive, flash memory or CD

[Home](#)

The view sector is a very useful function to get a feel for the state of a disk. A significant feature of the screen display is that for certain types of sector, the contents will be enhanced with a tool tip, ie a pop up message describing the information. The number of sectors that will be described will grow as the program develops.



The sector number is entered in box at the bottom of the screen. Any number may be entered, and the sector will be displayed as quickly as possible. To enter a number in Hex, prefix the number by 0x, eg 0x101 will display sector 257.

There are two options,

Cluster mode

In this mode, blocks will be read as clusters, depending on the operating system being used. Thus on a FAT disk, cluster 2 is the start of data section. The blocks are then shown as the cluster length, rather than the sector length.

Expand

Expand mode is for compressed NTFS disks. The data will be expanded, and displayed in expanded mode. The sectors are expanded unconditionally, and so a non compressed sector will expand to undefined results. However, a sector that has been compressed will expand

correctly, assuming the viewed sector is the start of the compression block.

Save...

The save functions allows a sector to be saved as a file. The files is a straight copy of the sector, so this is an easy way to dump hard drive sectors. If a complete dump of a drive or disk is required, then the [Image and raw](#) recovery mode should be used

Very slow reading

When a disk is failing, it is very common for a read to take a very long time (several seconds) before either coming up with a valid sector, or a failed sector. A sector displayed with just 5AH is a way that is used to indicate a failed sector. Occasionally, by jumping to a completely different location, and back, reading may be achieved quicker. Other approaches maybe to turn the drive off for a few minutes to cool down.

Copy function

The standard Windows copy function, Control C, can be used to make a copy of the sector contents. The area of the sector to be copied to the clip board must first be highlighted, it can then be saved in another document or e-mail with the Paste or Control V function. This can be useful if there is a problem that needs reporting.

Update

Occasionally errors can be seen that are due to incorrect values on a sector. Recovery could be possible, but only if the sector is changed. A CnW rule is that no sector on a disk must ever be changed, but this does not apply to a disk image. If the hex values are edited, then the sector can be saved to the disk image. Forensically, any change must be noted. There is always a question asking if the sector is to be updated, and this will enable the update to be aborted.

-0-

Disk imaging

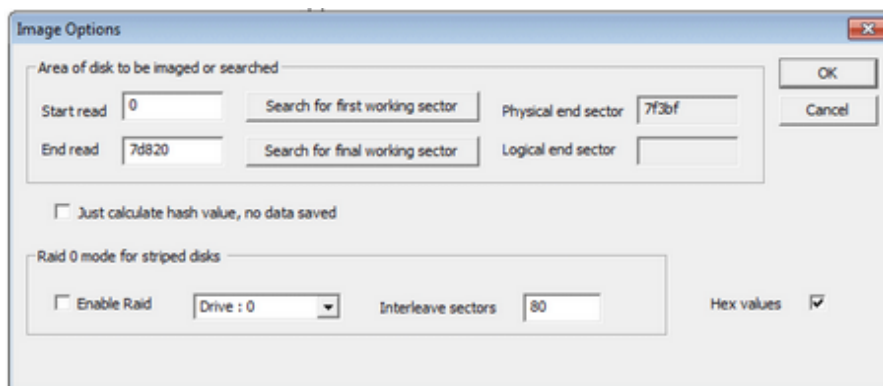
[Home](#)

This option is used to read the media at a very basic level from which either

- A complete image of the media may be made,
- Image can be built up incrementally

The image mode can either be as a copy of the disk for security reasons, or because the disk has a large number of sector errors. If there are a large number of sector errors, a disk image may be constructed in [sections](#), with the ability to skip bad areas of the disk. The file extraction tools are the same as recovering from any physical disk, with any operating system.

The image file created is in effect a DD format (as used in Unix/Linux). There is always a one to one mapping of each sector. For failed sectors, the image file is padded with 0x5A ('ZZZZ')



The displayed menu has a section where the media may be determined – which may help in calculating the correct size. The size of the scan – in sectors, may be entered by hand and this is typically used when one section of the media has become unreadable, for instance the start of the end of the media. By default, the program will try and read the whole disk.

The options allow extracting files in different ways, including allowance for compressed, or partially compressed NTFS disks

Raid Disks

When the Raid enable is selected, the image file will be produced in a striped format, allowing both sections of the raid to be logically placed. Full details see [Raid Disks](#)

Status

When doing a drive scan, status details are displayed. This will include the sectors read, and sectors failed. If the data is being split, it will also give the number of files discovered. The progress bar at the bottom of the screen will also indicate how far through the disk has been scanned. It will be discovered that sometimes the system appears to be extremely slow, and this is normally related to a high number of bad sectors. Unfortunately, there are not many ways over this problem, but the following suggestions may assist. It should be noted that it is always possible to cancel the copy, and the start again at the same location

- First suggestion is to cancel copy (but not current sector), and turn drive off. Once cooled down it may be possible to start again at the same location
- The second suggestion is to skip a section of the disk. Note where the copy has reached, and cancel copy. Start copy again, but make the start sector a high value. Write to the same output file, and the program will pad the apparent gap with sectors filled with 05AH. At any time later, it is always possible to follow a similar procedure, and start again in the area that was giving problems

On an NTFS disk that is reading very slowly, it may be worth while after reading the first few MB, to start the image at the start of the MFTs. This value is displayed in the [Recover option](#) menu for NTFS. Please read the [next section](#) on recovery methods

A curious observation of slow disk reading is that for some disks, it will sometimes suddenly speed up, and may even continue at high speed to the end of a disk. Patience can sometimes be rewarded with a lot of data, but at other times, no significant progress is made.

RAID disk recovery

[Home](#)

A RAID is a Redundant Array of Inexpensive Disks or sometimes Redundant Array of Individual Disks.

There are probably two main reasons behind a RAID, speed and security. There are also many variations of RAID giving different levels of security and speed.

CnW Recovery currently handles RAID 0 which is a striped set of 2 disks. In this mode data is written in blocks, such as 128 sectors to each disk in turn. The benefit can be increase in speed. The downside is there is no error recovery built in, so statistically a RAID 0 is twice as likely to fail as a normal drive. A failure of a single drive, then often means that the complete RAID has failed.

RAID 1 is a complete mirrored drive. ie two identical drives, and typically no speed benefit, but a single drive failure means no loss of data.

Higher RAID systems have a mixture of redundancy and speed benefits and the design goals are to get complete security from a single disk failure, but at the same time requiring fewer than double the number of drives, as in RAID 1. Data is therefore split over several drives in a way that any failure will allow all data to be reconstructed. This feature is only supported by the [RAID](#) option within CnW Recovery software

CnW Software handles RAID 0 Striped disks via the [Image disk function](#). An image file is created with the RAID option enabled. Both disks need to be read and it is important that it is determined which is Disk 0 and which is Disk 1. It is also important that the strip size is determined and set. Both disks are then imaged to the same output drive and data is placed in the correct logical location. A small problem with this approach may be the necessity to have a temporary drive with the capacity to store an image of both disk drives. Once the image has been created, it is treated as a simple disk image - the RAID element has in effect been removed

As with standard imaging, the image need not be done in a single go, thus areas of a bad disk may be skipped and data will still be placed in the correct logical location. Thus a RAID with a errors in one area on one drive can largely be recovered.

Interleave size

To recreate a logical image of the interleave is critical. Typical values are 0x80 or 0x100 (128 or 256) sectors. Determining the value does require looking at sectors to work out which physical location they should be in and

seeing it matches a possible interleave value. The choice of disk is much easier, disk 0 will always store the start of the disk, so expect to see a boot sector at the start of disk 0 and not at the start of disk 1.

For an NTFS disk, the MFT often starts at 0x60003F, so for a RAID0 this will be 0x30003F. It is then a matter of looking at the values of each MFT

-0-

How to use incremental imaging to recover damaged drives

[Home](#)

Very often a drive will partially work, or work very slowly. Thus areas of the drive will read, and others may have failed, or read extremely slowly. The solution is to use incremental imaging as described below.

A problem with a drive that has failed sectors is that attempts to read the physical drive are extremely slow. It could take days or weeks to read, or attempt to read all the sectors on such a drive. It is a commonly seen problem for 2.5" drive to read very slowly, often due to head wear. However, often the drive does read valid sectors after many retries.

The solution to the above speed problem is described below. It is fairly complex and does require some knowledge of disks, but can be a great help in recovering data in hours rather than weeks.

A key point to note when using an image file for a disk is that sectors can be missing, but every sector must be in the correct location. This means that failed sectors can be padded. When an image file is created, the user can select the start location, and the end location (in sectors) that are to be imaged from the drive. These sectors are then added to the image file in the correct location. If the sectors to be added are after the end of an existing image file, the file will be padded and then the sectors added. An example of this is there could be an image file of the first 1,000,000 sectors (500MB). If it is required to add an NTFS directory (MFT file) a typical location would be 0x60003F, or 6,291,519. Thus a read starting at 6,291,519 and ending 7,000,000 would read in the MFT file (assuming it is not fragmented and the data between 1 and 6MB would be padded).

How to determine where to read on a disk

There are several stages that should be followed to determine where a difficult to read disk should be imaged. It does also depend on the type of operating system and type of disk. The instructions below give guide lines for different operating systems.

One of the steps below the user will typically be swapping between the physical disk, and the image file

NTFS

NTFS is probably one of the easier types of disk to recover in this fragmented mode as the main directory is stored in the MFT file which is often a long, unfragmented file. CnW Recovery can also recover files with the necessity of using the Index files.

1. The first stage is to determine the location of the MFT. As long as the boot sector can be read then an indication of the start of the NTFS partition will be seen. For a single partition drive this is often sector 63 (0x3f). This sector will then indicate the location of the MFT. So stage one is to read the boot sector (sector 0) and the first few sectors at the start of the partition.
2. The second stage is to run the Recover function (using the image file) and then see the location of start of the MFT. One then needs to image from the start of the MFT for it's possible length. Each entry is 1024 bytes long, so normally 2 sectors. Therefore, if the MFT entries field looks valid, one will need to read twice the number of sectors
3. The third stage is to determine where the files for recovery are. This is done using the recover function, and Recover from File entries. Make sure that you select the 'Select Files' function. At this point the program will scan the disk for all files and most importantly create a log entry. The log can then be used to determine where a specific file is stored, or a group of required files is stored. It maybe that required files are stored in the 30GB area. This region of the disk can then be imaged.
4. The final stage is to repeat stage 3, but this time, one the directory has been display, select the files to be recovered, and recover them.

The advantage of using the above sequence is that a failing sector is only read once, and all sectors in the image file will be read at high speed, irrespective of whether the sector is good or bad. An image may then be built up to contain just the areas of required files. A real life example of this technique was a 60GB disk that imaged upto about 30 GB, and then went very slow. By using stage 3 above, it was possible to determine that only a few areas beyond 30Gb were required and these could be targetted, and large areas safely omitted.

FAT

On a FAT disk, directories are stored in all areas of the disk. This has the advantage that a failure of one area of the disk will not necessarily kill the complete directory and file information, as could happen with NTFS, or HFS+. It does make looking for directory areas much harder.

MAC

As with NTFS, the MAC HFS+ does store directory entries in a file and the recover options menu will give locations of this catalog file. The important part is to recover enough of the data at the start of the disk to display the catalog file

Disks with single head failure

[Home](#)

Modern disk drives have multiple platters and hence multiple heads. Sometimes a single head can fail that means areas of the disk will read correctly, but other areas will either fail totally, or read extremely slowly, typically with lots of errors. A drive head replacement may help in these circumstances, but this does require specialist hard drive repair facilities. If a partial recovery is acceptable, or the cost of head replacement too expensive, CnW Recovery will allow imaging of the disk drive in a way that the failing head can be ignored.

The option is set up in the Configuration menu so that bad sectors when detected are skipped, but skipped with a jump. It is worth doing some playing around with the drive to try and determine the size of data to be skipped caused by a bad head. On one drive, a 500GB drive, there appear to be 4 heads, and the length of each track was about 192,500 sectors. To create a disk image the program was set to skip errors in on a single failure, and then skip 12,050 sectors. The skip value is not too important. A small value will be slower, but a large value could lead to more good data being skipped as well. This value can be changed at any time during the imaging process by selecting the Configure Icon.

The resulting image will not be 100% complete, but data can be recovered by software means only.

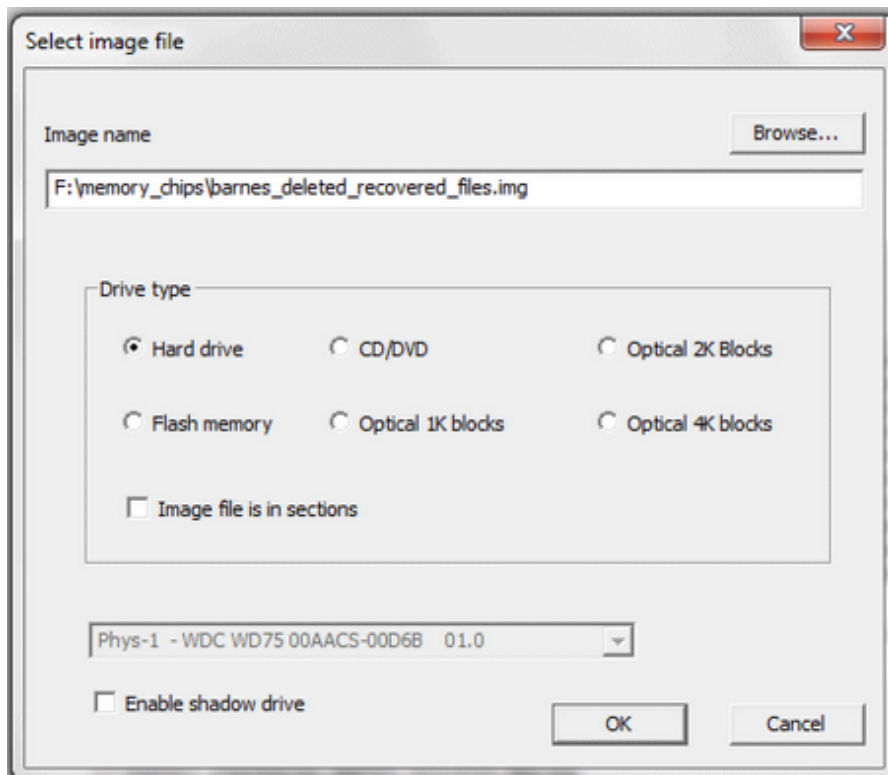
-o-

Image file selection

[Home](#)

Cnw Recovery will work with either a physical hard drive, or with a DD image file. This is a file where there is a one to one mapping of each sector to the image file. To select the image file, use the drop down drive select box at the top of the screen and select 1: Image file and Backup files.

The screen below will be displayed



The browse function allows an image file to be selected. This can either be a DD type image, or a supported backup file, such as a Microsoft MTF file

The drive type actually selects the block size. For a hard disk, the block size is 512 bytes (or 0x200). A CD and DVD have 2048 byte blocks. Optical disks come in several variations, which may be selected.

The Image file is in sections (not yet implemented) will allow for systems that generate files, typically in DVD size sections

Shadow Drive

The shadow drive is a useful feature when a disk has only been partially imaged. This may be the case for a disk with many failures. If this option is enabled, a physical drive can be set as a shadow drive. When the disk image

is read, and the sector is determined as unread, or failed, the program will try and read the shadow drive. If successful, the disk image will be updated, If unsuccessful, due to a total sector failure, the disk image will be marked to indicate that the sector can not be read. This process ensures that the drive is not worn out by many sector retries.

Forensic Options

For forensic packages, two other file types will be recognised and processed.

Virtual Disk Format

This is an image format built up in sections - referred to as Grains. The basic image is a sparse image, so only allocated sectors are saved.

Encase E01

The E01 format is a commonly used name for EWF format (Expert Witness Compression Format). It has been adopted by Encase and is a standard forensic format. It consists of one or multiple files, with or without compression. As part of the format, each section has its own MD5 hash value and so is very secure and any corruption in storage will be detected.

Encrypted Drive

This option should be set to read certain WD encrypted drives - (Forensic only option)

-0-

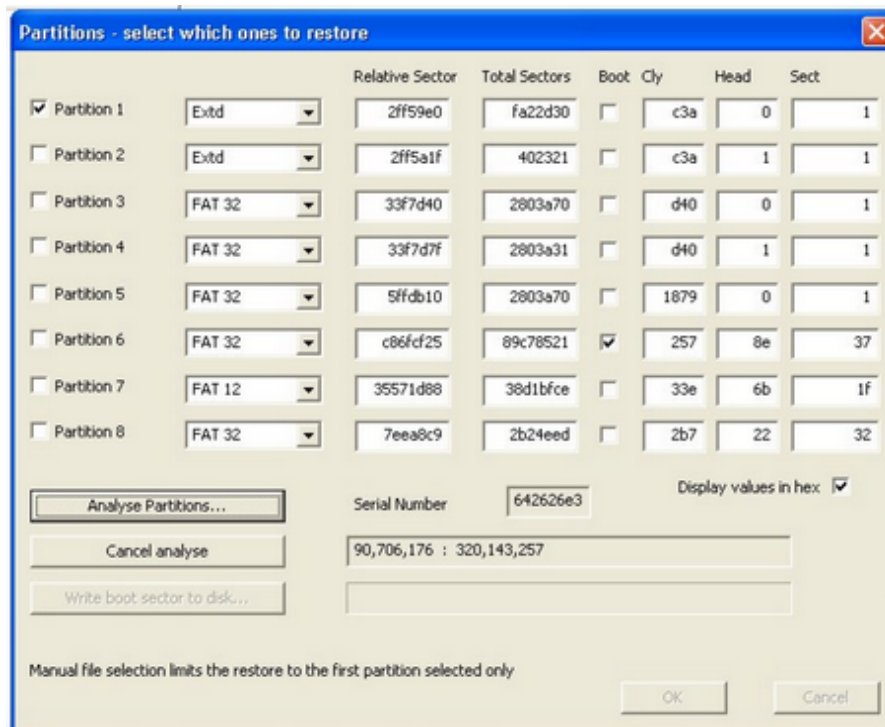
Partitions, analysis and recovery

[Home](#)

Probably the most important sector on a hard disk is the boot sector, or also called Master Boot Record (MBR). If this sector fails, or is corrupted, most PCs will not read the disk at all. Thus on a disk, or memory chip where the boot sector is invalid, when doing a recover, the error message "First partition not recognised, Run the analyse partitions function" will be displayed, and the dialog box below will be displayed with ??? rather than a format type.

When running the Recover function, if a disk has more than one partition, an option will be displayed so that only the required partitions are restored. Each partition – up to 8, will be displayed with detected format, start and end sector on the disk. The values displayed are taken from the boot sector (see below). If the boot sector is corrupted, new values can be entered. Alternatively, the Analyse Partitions function can be run. This will scan the whole disk, looking for possible partition starts.

A very significant feature of the CnW Recovery software is that it is not necessary to write a new partition sector back to the disk in order to recover data. Once parameters are edited, a temporary copy of the boot sector in memory is used. This means that the master disk is not changed (essential for forensic investigation) and if the disk has a completely failed sector 0, this does not cause a problem.



The display shows data about the disk as follows

- Operating system. This is as read in the partition table, and can be NTFS, FAT32 etc
- Relative Sector. This the start of the logical partition. Often it is sector 63
- Total sectors. This is the number of sectors in the partition. For many NTFS partitions there is an extra sector at the end, a copy of the parameter sector
- Boot. This indicates that the marked partition is a bootable partition
- Cly – this is the cylinder number. As the actual parameter table has a limit of 1023 for this value, CnW program displays the logical value
- Head. This is the value of the head for the start of the associated partition. The value will be 0-254
- Sector. This is the first sector in the track for the partition, it is normally 1, and has a maximum of 63

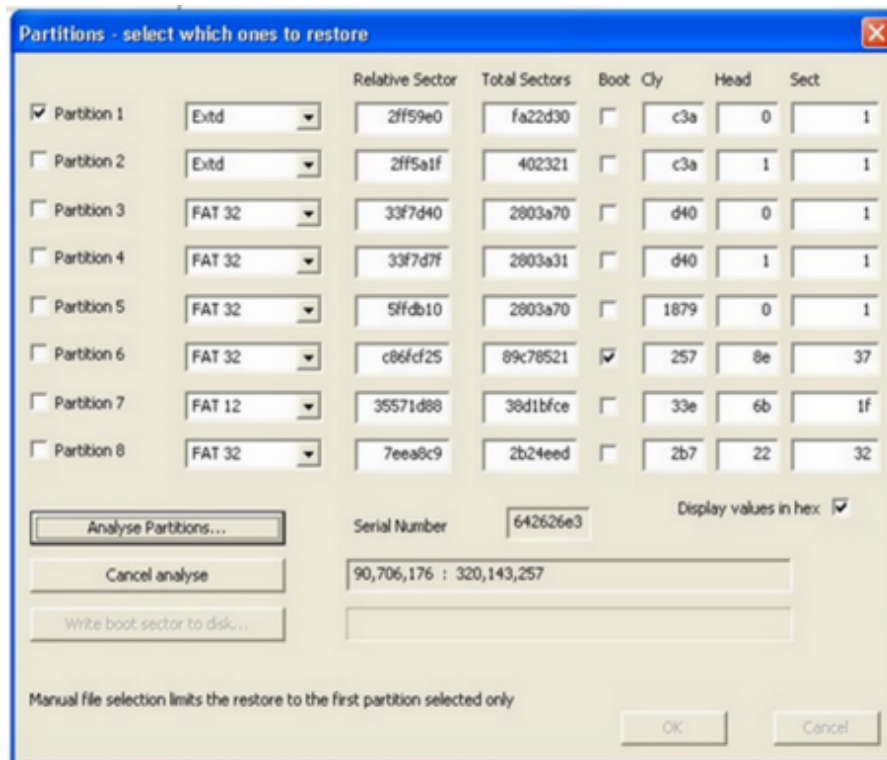
The above values may be changed, but for recovery purposes, only the Relative Sector, and Total sectors are used. If the relative sector is wrong, there will be problems restoring data. The Total sector count is not as important, and if too large will probably not affect the program.

When the analyse function is run, it will try and reconstruct the values described above. Once the analyse has been run, it is possible to do a test restore using the newly recovered values. It is not necessary at this stage to write the new sector to the hard disk. This also means that the demo program will work with a new boot sector layout.

If you multiply the maximum cylinder, head and sector numbers together, and then the sector size of 512 bytes, you get to 8GB. Some users may be familiar with this being a limit some years of ago the maximum disk size that PCs could accept.

Boot Sector or Master Boot Record

The boot sector is sector 0 on a hard drive, and a typical screen dump of one is shown below



There are several areas to look at on the dump.

Most of the sector, from the start to byte 0x1bd is code that is run to start the PC up. This is only important for a bootable disk, and can be ignored for straight forward data recovery. If it is required to re-instate the booting of the disk, this code must be valid.

The final two bytes of the sector are always 0x55 and 0xAA. These are check bytes that help ensure that the sector has been read correctly. For instance, it is occasionally possible for a byte to be skipped, or read twice. In these cases, the 0x55 and 0xAA would not be in the correct location. The hard drive CRC check should also fail, but that is much harder to see - though the PC should detect an error.

The important part of the disk starts at byte 0x1BE, and it is these bytes that are decoded by CnW program and displayed logically in the table at the top of this page. The brief summary below will describe each byte. There are up to 4 entries, each 16 bytes long, for each partition. If a partition is marked as extended, then the pointer points to another sector with the same data structure starting at 0x1BE on the new sector. This is a means where an unlimited number of partitions can be created. See Partition Table Structure for full details

-0-

Partition analysis mode

[Home](#)

There are two modes to try and analyse the partitions on a disk. One is to recover the current partitions. The second is to try and recover a previous setup of partitions on a disk that has been repartitioned.

Reconstruct current partitions

This is the quick mode. The program will start scanning the start of the drive until it finds a media partition sector. At this point it will try and follow the partitions through the disk until the end is reached. If the partitions do not chain, then the program will continue scanning every sector - this is obviously slow.

If no partitions are found, the program does try and detect the type of operating system on the disk. For instance NTFS and FAT disks will be detected.

Search for previous partitions

This is mode where the whole disk is scanned for possible partition starts. Obviously this can be slow, but find all possible partitions. Each possible partition start will be analysed but only ones that point to a valid FAT start or NTFS entry will be displayed. This way most false entries will be ignored

Stop searching when first partition found

Many disks are known to have only a single partition. If this is the case, then checking this box may save a considerable period of time, preventing the program searching for other possible partitions.

Test for partition

There are flags that must be set to indicate which partitions are to be tested for. Thus if only searching for a NTFS partition, but setting just the NTFS flag, no FAT partitions will be detected. There must always be one partition type set, but any combination can be used.

As the program scans through the disk, the partition fields will be updated. If it is felt that valid information has been added, then the scanning can be cancelled and values found to date will be used. Often, partition information is only at the start of a disk, so there is little requirement to scan a complete 500GB disk

Partition Table structure

[Home](#)

A partition table is a structure of 16 bytes. There can be up to 4 tables in a boot sector, and the first record always starts at location 0x1BE. An example is shown below.

```
0001B0  00 00 00 00 00 00 00 00 00 - CA EE BA 36 00 00 00 01  01°6
0001C0  01 00 07 FE 7F D7 3F 00 - 00 00 99 FF 14 13 00 00  b*x?  0000
```

Each table is the same structure - or may be blank

Location	Description
----------	-------------

0x0	0x80, this partition is the boot partition, 0x00 not bootable
-----	---

0x1	Address of first cylinder
-----	---------------------------

0x2	Address of first head
-----	-----------------------

0x3	Address of first sector
-----	-------------------------

0x4	Partition type. This can have many values, but the list below represent the most common values
-----	--

0x00	Unused - means this partition table is not used
------	---

0x01	FAT 12
------	--------

0x04	FAT 16 - upto 32MB
------	--------------------

0x05	
------	--

0x0f	Extended partition. This will point to a new sector, acting like a MBR
------	--

0x06	
------	--

0x0e	
------	--

0xde	FAT 16
------	--------

0x07	NTFS
------	------

0x0b	
------	--

0x0c	FAT32
------	-------

0x1b	
------	--

0x1c	Hidden FAT32
------	--------------

0x63	Unix SCO
------	----------

0xa8
0xab Apple Macintosh

For more values

0x5 Address of last cylinder - often not valid for large disks

0x6 Address of last head - often not valid for large disks

0x7 Address of last sector - often not valid for large disks

0x8-0xb LBA of first sector in partition. Will point to a [Parameter block](#). 0x3f is a very typical value

0xc-0xf LBA of final sector in partition. For a single partition disk this will normally be the end of the disk

-0-

How to recover corrupted partitions

[Home](#)

At first glance recovering a disk with a missing or corrupted partition table can be a bit daunting. The steps below will assist.

It is very useful to know how the disk was partitioned - though obviously this information is not always available.

Typical setups are often as follows

- Single partition of complete drive in FAT32 or NTFS
- Dual partition of a drive, normally FAT or NTFS
- A drive that has a hidden recovery partition. The recovery partition is often FAT16 or FAT32
- A drive with extended partitions, and more than 4 basic partitions
- A drive that has been re-partitioned with similar, or very different parameters
- A drive that 'crashed' when running a repartitioning program

The complexity increases as one goes down the list

Partitions can be located in a few different ways

- By Master Boot record
- By media BIOS
- By finding start of MFT for NTFS, or subdirectories for FAT disks
- By hand

On a good disk, the Master Boot Record (MBR) contains a table, starting at location 0x1BE. This will contain information on upto 4 partitions. When more than 4 partitions are required, one or more of the pointers will be for an extended partition. In theory the number of partitions can be unlimited - CnW recovery handles the first 8 automatically.

Missing MBR (Master Boot Record)

When the boot sector (sector 0) is missing, or totally corrupted, the first approach to try is the [Analyse Partitions](#) function. Reconstruct current partitions will search for the first media bios record, and then try and scan through the disk from there. Search for previous partitions will scan the whole disk, but will detect any possible media bios sectors. If just the MBR has failed, then reconstruct current partitions will work. If the disk has failed while being processed, such a repartitioning program, it may be best to use the Search for previous partitions.

After analyse is run, or if done manually, the partition table must have the following information, the partition type, eg NTFS, FAT16 and the Relative sector, which is the start sector of the media BIOS. The length of the partition is not critical, and if in doubt make the number too large rather than too small.

Once the values for the boot sector have been determined, it is possible to write them back to the boot sector. However, this is not actually required for the recovery process. The program will remember the values and allow the user to do a recovery, or a trial recovery without making any physical changes to the disk. For forensic applications, this is extremely valuable. For disks where sector zero has failed, it is not necessary to have a working sector zero.

-0-

GUID Partition tables

[Home](#)

The standard boot sector is limited in the fact that it only has direct support for 4 partitions. Extra partitions are added by chaining new partition tables. It works, but is rather messy.

A new standard is the Extensible Firmware Interface (EFI) an Intel replacement for the Master Boot Record (MBR) that is still on PCs. Current Apple OS/X systems use this new partition layout. It is an option on Vista, and essential to create partitions that exceed to TB. The existing boot sector only has space for 32 bit sector numbers, the new system handles 64 bit sector numbers - should be a few years before that becomes a limitation.

The structure is that the MBR looks like a normal boot sector, with a single partition entry. The critical point is the the file system is defined as 0xEE, rather than say NTFS, or FAT16. The first sector is marked by the string "EFI PART". The following sectors contain the specific partitions with a 128 byte record for each partition.

Each partition entry is identified by 2 16 byte GUID (Globally unique identifier). It then contains both start and end sector number, and a text description of the partition such as "Apple_HFS_Untitled_1"

CnW Recovery software reads this type of header and determines the disk type. The program currently supports Apple HFS+ partitions and Windows Data partitions, as may be found on Vista drives. See the section on recognising sector types for more details

-0-

Magnetic Media Recognition

[Home](#)

To aid with recovery of a disk, it is often worth knowing what type of operating system has been used. For a CD / DVD they are normally either ISO9660 or UDF. Most memory chips are FAT (all variation, FAT12, FAT16 and FAT32). While a hard drive is normally either FAT32 or NTFS or Linux

If an optical disk is being read, they can be NTFS, FAT, HPOFS or one of many proprietary formats which is beyond the scope of CnW Recovery Software, though a Raw recovery may assist.

Very often, CnW Software will automatically detect the relevant format, but there can be times when for instance a disk has been reformatted with a new operating system. In this case, it is often useful to scan through the disk to find sectors that indicate exactly what the format was.

With a good disk, the location to start is with the boot sector, and this is decoded by the [Partition](#) function. However, this function cannot immediately display partitions that have been overwritten, or totally corrupted, a more manual process is required.

-0-

Deleted file recovery

[Home](#)

A very common problem with all computer storage is accidental deletion of files.

On most operating systems, when a file is deleted the directory entry is marked as deleted, and the space is made available for re-use. If nothing else is written to the disk, then there is a very good chance of recovering the files. One analogy would be if a telephone directory was torn up, it would still be possible to telephone the existing numbers.

There are two major considerations on deleted file recovery and these are overwritten files, and fragmented files.

Overwritten files

When a file is deleted, the space is made available for new files. Any new file could therefore use the space that a previous file - now deleted - used. Depending on the file, this could render the file totally unusable, or just a section with unknown data in it. CnW Recovery tries to detect when a file has been potentially overwritten, and recovered files will be stored in an 'Overwritten' directory.

Fragmented files

The optimum way for any operating system to write a file is as a continuous stream. As the disk gets full, or very large file are written, it is often necessary to write a file in several sections or fragments, and hence we get fragmented files. Recovery is therefore more complex.

A FAT drive stores the start of a file in the directory, but then each cluster location is stored in the File Allocation table. This table is cleared down when a file is deleted, and so no record of how the file was stored is kept. On recovery, it is often only possible to assume that the file was sequential, and on many occasions, this leads to a good recovery. On a very large file, or very full disk, the success rate drops.

NTFS disks have a major advantage over FAT disks described above in that often the first 10 or so fragments are stored in the directory, or Master File Table (MFT) block. Thus a partially fragmented file can be recovered without errors. The structure of an MFT is very complex, and in some cases and MFT may comprise of more than 10 separate MFTs, so very fragmented files can still represent major problems.

Ext4 disks - it is not possible to recover deleted file except by data carving. All metadata is cleared when the file is deleted.

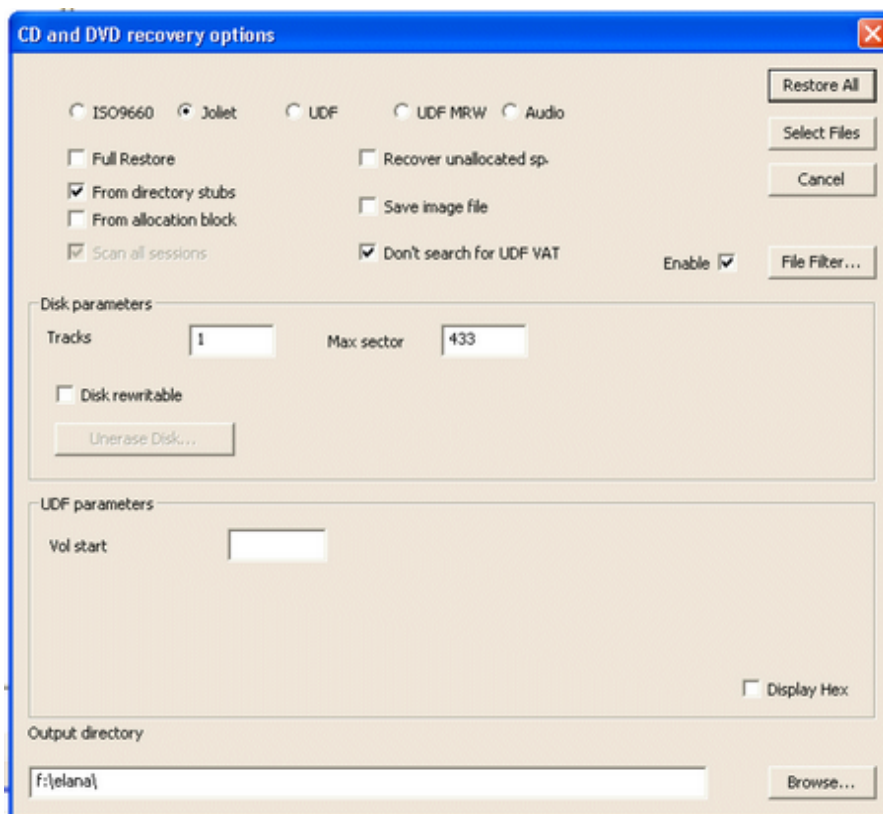
-0-

CD Recovery

[Home](#)

A CD may be restored in several ways, depending on if it is working correctly, or has failed in one way or another.

The program will try and determine the type of disk being read, but this can be over written by selecting 9660, Joliet, UDF, UDF with Mount Rainier, or Audio disks



There are two basic modes of restoration, Full Restore and From directory stubs

Full Restore

In Full Restore mode the program reads the disk in the a same way as an application would, but is very tolerant of errors. Files will be restored, and save in original subdirectories. For a multi-session disk, each session is saved in a separate base directory, Track1, Track2 etc. This has a feature where if a disk has been appended to, say 10 times, a file in Track1 may appear in every other track

From directory stubs

This is more of a recovery mode. The disk is scanned for sectors that are directories. Depending on the disk, the scanning can take a period

of time, but the progress bar will indicate the amount of the disk scanned. The information is then decoded, and files recovered. Where ever possible, the parents of subdirectories are determined. Where no parent can be found, the found, the files are placed in a dirstub0, dirstub1 directory.

Scan all sessions

This is an option for UDF disks that have been written with packets, rather than separate sessions, or tracks. (It is also part of the forensic options, so not available on all versions of the software). It will show each session as a separate track. This way, each session could be restored separately, but more importantly, for a forensic investigation, one can see when files are added, or deleted. A normal restoration of the disk, will only restore the final session. One warning is that this examination process is rather slow, as the disk directory is constructed many times. This can also be a useful option when the final part of the disk is missing or corrupted. All sessions, up to the final one can then be recovered.

Disk Parameters

This section shows the number of tracks detected on the disk, along with the maximum sector number. As each sector is 2K in length, this will give the maximum number of bytes that can be restored. It should be noted though, that on multi [session](#) disks, a lot of data space is used as overhead between sessions, so the capacity of a disk with many tracks, if recorded in different sessions, will be much lower than the Max sector x 2K

Unerase disk

When the program detects a blank CD-RW disk, the unerase option is enabled. If a disk has been Quick Erased, there is a procedure where data may be recovered - see [Unerase CD-RW](#) for more details. This is Forensics only option, and so may not be enabled on all versions of the software licence.

-o-

How to recognise type of CD/DVD

[Home](#)

With a very corrupted CD or DVD it may be necessary to set the type of format manually. It is therefore essential to know how each format is recognised. Fortunately, CnW Recovery software will normally determine the disk correctly, but problems can occur when a significant part of the disk (typically the start) has been made unreadable.

There are two common formats for CDs and DVDs. These are ISO9660 / Joliet and UDF. DVDs tend to be UDF, but this is not always the case. There is also a middle case where a disk is both ISO9660 and UDF. On these disks, the data is stored once, but there are two parallel directory structures pointing to the same files.

ISO9660 / Joliet

The first location to look on a CD is sector 16 (10H). If the sector contains the string "CD001" then this is a ISO9660. A joliet disk is very similar, and typically, sector 17 (11H) will have the string "CD001", but the volume name in byte offset 1aH will be double spaced uni-code

Sector 16 for ISO9660

```

000000  01 43 44 30 30 31 01 00 - 20 20 20 20 20 20 20 20  CD001
000010  20 20 20 20 20 20 20 20 - 20 20 20 20 20 20 20 20
000020  20 20 20 20 20 20 20 20 - 42 41 43 4B 5F 55 50 5F  BACK_UP_
000030  44 56 44 5F 31 20 20 20 - 20 20 20 20 20 20 20 20  DVD_1
000040  20 20 20 20 20 20 20 20 - 00 00 00 00 00 00 00 00
000050  D0 3B 17 00 00 17 3B D0 - 00 00 00 00 00 00 00 00  ð; ;ð
000060  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000070  00 00 00 00 00 00 00 00 - 01 00 00 01 01 00 00 01
000080  00 08 08 00 F2 15 00 00 - 00 00 15 F2 A8 01 00 00  ò  ò"
000090  00 00 00 00 00 00 01 AB - 00 00 00 00 22 00 13 00  «  "
0000A0  00 00 00 00 00 13 00 08 - 00 00 00 00 08 00 6B 03  k
0000B0  15 14 38 00 00 02 00 00 - 01 00 00 01 01 00 20 20  8
0000C0  20 20 20 20 20 20 20 20 - 20 20 20 20 20 20 20 20
0000D0  20 20 20 20 20 20 20 20 - 20 20 20 20 20 20 20 20
0000E0  20 20 20 20 20 20 20 20 - 20 20 20 20 20 20 20 20
0000F0  20 20 20 20 20 20 20 20 - 20 20 20 20 20 20 20 20
000100  20 20 20 20 20 20 20 20 - 20 20 20 20 20 20 20 20

```

Sector 17 - Joliet

```

000000  02 43 44 30 30 31 01 00 - 00 20 00 20 00 20 00 20  CD001
000010  00 20 00 20 00 20 00 20 - 00 20 00 20 00 20 00 20
000020  00 20 00 20 00 20 00 00 - 00 42 00 61 00 63 00 6B  B a c k
000030  00 2D 00 75 00 70 00 20 - 00 44 00 56 00 44 00 20  - u p  D V D
000040  00 31 00 00 00 20 00 00 - 00 00 00 00 00 00 00 00  1
000050  D0 3B 17 00 00 17 3B D0 - 25 2F 45 00 00 00 00 00  ð; ;ð%/E
000060  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000070  00 00 00 00 00 00 00 00 - 01 00 00 01 01 00 00 01
000080  00 08 08 00 1A 4C 00 00 - 00 00 4C 1A 66 03 00 00  L  Lf
000090  00 00 00 00 00 00 03 70 - 00 00 00 00 22 00 AE 01  p  " ®
0000A0  00 00 00 00 01 AE 00 08 - 00 00 00 00 08 00 6B 03  ®  k
0000B0  15 14 38 00 00 02 00 00 - 01 00 00 01 01 00 00 20  8
0000C0  00 20 00 20 00 20 00 20 - 00 20 00 20 00 20 00 20
0000D0  00 20 00 20 00 20 00 20 - 00 20 00 20 00 20 00 20
0000E0  00 20 00 20 00 20 00 20 - 00 20 00 20 00 20 00 20

```



```

0000F0  00 20 00 20 00 20 00 20 - 00 20 00 20 00 20 00 20
000100  00 20 00 20 00 20 00 20 - 00 20 00 20 00 20 00 20

```

UDF

A UDF disk has several very distinctive features that can be looked for. Often the first two bytes are the most relevant values on a control sector. The sectors with tags 1-9 below all follow the Primary Vol descriptor. The sequence is not important, but the final sector in the chain is always tag 8.

tag 1 (ECMA167 3/10.1) Primary Vol descriptor	0x01	0x00
tag 2 Anchor Vol Pointer normally at location 256	0x02	0x00
tag 3 Volume descriptor pointer	0x03	0x00
tag 4 Implementation use volume descriptor	0x04	0x00
tag 5 Partition descriptor	0x05	0x00
tag 6 Logical volume descriptor	0x06	0x00
tag 7 Unallocated space descriptor	0x07	0x00
tag 8 Terminating descriptor	0x08	0x00
tag 9 Logical vol integrity descriptor	0x09	0x00
tag 256 Fileset descriptor	0x00	0x01
tag 257 File identifier descriptor	0x01	0x01
tag 258 Allocation length descriptor	0x02	0x01
tag 259 Indirect entry	0x03	0x01
tag 260 Terminal entry	0x04	0x01
tag 261 File entry	0x05	0x01
tag 262 Extended attribute header descriptor	0x06	0x01
tag 263 Unallocated space descriptor	0x07	0x01
tag 264 Space bitmap descriptor	0x08	0x01
tag 265 Partition integrity entry	0x09	0x01
tag 266 Extended file entry	0x0a	0x01

The first three sectors of a UDF disk - often after the Joilet sectors for a Bridge disk

```

000000  00 42 45 41 30 31 01 00 - 00 00 00 00 00 00 00 00  BEA01
000010  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000020  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

000000  00 4E 53 52 30 32 01 00 - 00 00 00 00 00 00 00 00  NSR02
000010  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000020  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

000000  00 54 45 41 30 31 01 00 - 00 00 00 00 00 00 00 00  TEA01
000010  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000020  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

Anchor sector 0x100

```

000000  02 00 02 00 CE 00 00 00 - 01 D7 F0 01 00 01 00 00  î   xä
000010  00 80 00 00 20 00 00 00 - 00 80 00 00 30 00 00 00  e   e 0
000020  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

Bridge Disk

A bridge disk has both UDF and ISO9660 structures.

UDF Anchor Volume

[Home](#)

To read a UDF disk logically, several critical sectors must be read, that then set pointers to the next critical sector. The first such sector is the Anchor Volume Descriptor. The documentation states that there should be at least 2 such anchor blocks in three possible locations

- At sector 256 (100h)
- At the final sector of the disk
- 256 blocks before the final sector of the disk

It is quite possible to have a recoverable CD or DVD with no valid Anchor block, and so manual intervention is required. As the Anchor block only contains information on where the Main Volume Descriptor is stored, all that needs to be entered is the location, and length of the main volume descriptor. Alternatively, the Search function can be used, this may be slow (hence the cancel function).

Main Volume Descriptor

The main volume descriptor is a sector that starts with the hex bytes 01 00 and the bytes in offset 12-15 is the sector number in little [endian](#) format

```

000000 01 00 02 00 DD 00 01 00 - F2 E3 F0 01 11 02 00 00  r 7 Y r 0E3r4
000010 00 00 00 00 00 00 00 00 - 10 00 52 00 6F 00 78 00  + R o x
000020 69 00 6F 00 00 00 00 00 - 00 00 00 00 00 00 00 00  i o
000030 00 00 00 00 00 00 00 00 - 01 00 01 00 02 00 03 00  r r 7 L
000040 01 00 00 00 01 00 00 00 - 08 30 46 42 38 38 42 46  r r 0FB8BF
000050 38 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00  8
000060 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000070 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000080 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000090 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000A0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000B0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000C0 00 00 00 00 00 00 00 09 - 00 4F 53 54 41 20 43 6F  OSTA Co
0000D0 6D 70 72 65 73 73 65 64 - 20 55 6E 69 63 6F 64 65  mpressed Unicode

```

Typical pointers to look are the name in offset 0x1a, and OSTA strings.

If the search function does not find the relevant sector, it may be worth while looking through sectors by hand. Typically it will be in a location in the low 200H, or may be 40H

If it is not possible to find a main volume descriptor, then it is likely that other critical sectors will also be missing.

-o-

Unerase CD-RW

[Home](#)

With CD-RW disks it is possible to erase them. Once erased, logically they look like a new blank disk. Physically though, one can normally see that the disk has been written to by the different colour on the data side of the disk. This is a forensics only option.

It must be noted that when dealing with blank disks, the program can take a long time (maybe 30 seconds) to try and read a blank sector. This can cause the program to appear to hang, so please do not give up immediately.

Most programs will treat an erased disk as just that, and there is no way to recover the data. CnW software does have a way that normally works to in effect unerase the disk, and make the disk readable again.

There are two modes to disk erasing, a quick erase, and a full erase. A quick erase just sets all pointers on the disk, in hidden areas to make it look like a new disk. This mode can be recovered. A full erase will overwrite each sector, and this means that data can never be recovered.

This routine changes elements of the disk so that it may read using CnW Recovery software. A small side effect is that the first 16 sectors of the disk remain blank, and it cannot be read using the standard operating system.

Recovery of erased CD-RW disks

A problem with an erased CD-RW is that it looks just like a blank disk, and it is impossible to determine what has happened to it without doing the unerase process. It must be noted that the unerase used in CnW does actually write to the disk and so potentially it could damage a disk that has failed, but not due to being erased. It should therefore be treated with caution, but at the same time it may be the only option. This function, believed to be unique on commercial user software, can be slightly problematic, but it does normally produce the desired results, although occasionally multiple attempts may be required.

The Wizard on CnW will highlight disks that are possibly erased. In these cases, the Manual Recovery mode should be entered, and on the [CD recovery options](#) menu there is a button for Unerase disk, which is only enabled when a blank disk is detected. When the function is selected, a new screen is displayed with many options on. As mentioned above, the configuration of the disk before erasing can normally only be guessed at, and so the unerase may need to be done in different ways until the correct combination is found.

It is necessary to try and replicate how the disk was written first of all. Some combinations will display the error message "Error setting SelectMode" when the Unerase button is pressed.

A good starting point is to use the variables shown above. The one very important parameter is the Data Block Type. If the incorrect value is chosen, then the data on the disk will be 8 bytes out of sync. However, there is a test at the end of the unerase that will ensure the correct value has been entered. If the value was wrong, then it will be necessary to Blank CD and select a different Data block type before doing a new Unerase function.

The result after Unerasing should be a message "Unerase complete and sector read OK". This shows that the program can now read a test sector after the unerase operation. The Normal CnW CD recover functions can now be used, but the user should be aware that it is occasional very slow starting.

Sometimes, there is a 'false' error message saying that the unerase has failed. Before doing anything, the disk should be tried, with the view function and an attempt to read sectors above 16 (0x10) should be made. If this works, then a recover function can be tried.

The parameters that can be changed are displayed below. As details of the original disk may be unknown, there is an element of trial and error

Fixed packet size

This mode is used on disks that have setup to be a read/write disk. For disks that are written as a standard CD, then the fixed packet size will not be set.

Packet size

The normal value is 0x20 (32 sectors). If fixed packet size is not set, this value is not used

Multi-session

This is a flag used for data will be added to the disk - it should be set to 3

Data block type

This is the type of disk that has been written. All three modes will be found on disks.

Write type

Session format

The most common type of disk will be a CD-DA

Error messages

Last RZone not visible. If this is found, re-blank the disk and try again

This program is part of the Forensic Option. If it is required to recover a CD-RW (or DVD-RW) then CnW do offer a service at a fixed fee of £30 (or £40 for DVD). Please e-mail info@cnwrecovery.com for more details.

The same software option for DVD-RW has not been developed yet, so the only solution is the service described above.

-0-

Multi-session UDF

[Home](#)

With UDF disks, files may be added, or deleted even on CD-R and DVD-R disks. The process is achieved by a Virtual Partition, controlled by a Virtual Allocation Table. With a CD or DVD as sector can never be changed but new sectors can be added. A disk contains a mixture of data area, and directory files. The virtual partition make use of the normal directory files, but they are access through a lookup table. This table is updated for when a new group of files is added or changed. Logically the reading program thinks it is reading logical sector 'x' but the look up table mans that this sector can in effect be updated by pointing logical sector 'x' to a new physical location.

When reading a UDF disk, the first stage the reading program has to do is to find the current look up table, or VAT. This is pointed to by the last sector written on the disk, and so disk searching starts from the end. An interesting feature of this mode of operation is that by searching through the disk for these VAT pointers, the state of the disk after each session can be determined. Forensically it is possible to see which files have been added, deleted or changed.

-0-

Camcorder Recovery

[Home](#)

Mini DVDs are typically used in Camcorders, Video cameras etc. The typical failure mode is related to sessions not closing, or just general failure at the start of disk.

Different types of camera do use slightly different logical recording methods, but fortunately, the basic standard is to record mpeg files, with some control files, IFO files. It is very common for a failed DVD not to have any of these IFO files, but recovery is still possible.

There are three possible ways to work on a Mini DVD that has failed

- Wizard
- Logical Recovery
- Raw Recovery

A mini DVD is normally recorded as an ISO9660 structure, or UDF with groups of files with the following extensions, IFO, BUP and VOB. The file IFO and BUP are identical. The VOB file stores the mpeg data and therefore is the important one to recover.

A VOB file is basically an MPEG file with addition information taken from the IFO and BUP files. The maximum size for a VOB is 1GB, and so on a long movie there will be multiple VOBs and matching IFO files. In addition there should also be a VIDEO_TS.IFO and VIDEO_TS.BUP, with an optional VIDEO_TS.VOB if there is a start menu. If it is possible to recover all of these files, then a new video disk can be created. If only the MPEG can be recovered, it is necessary to rebuild the IFO / BUP files. This is performed by a feature (currently under development) to [rebuild video files](#)

Wizard recovery

For many camcorder disks, they will be detected by the wizard as Corrupted Video Disks. If the screen indicates that several files are present, then a full recovery may work. Typically, it will be necessary to recover files from unallocated space.

Logical recovery

The Recover function will allow recovery of files in a logical way - as long as the disk has the basic control blocks still intact. If this fails, then Raw recovery will be the best option.

Raw recovery

The raw recovery mode is probably the most common mode for recovery of video disks. It will scan the complete disk, and extract

either a single large MPEG file, or many smaller mpegs, based on individual chapters. If the individual chapters are required, then the Separate video file chapters option should be selected. Different camcorders work in different way, so it may be best to try a recovery with 'Separate video file chapters' enabled and disabled.

Raw recovery can be done with a complete scan, but often the start of the disk cannot be read. It is therefore advantageous to determine where the data starts. There are two ways, one is to use view sector, and try different starting points, eg 20000, then 10000 and try and find the start by trial and error. The easier way is to use the built in function, Search for start sector. Once the start location is determined, and the options set (Split on possible file starts and Separate video file chapters) then a scan can be performed.

If after a period of time the scan moves very slowly, and comes up with a significant number of errors, the scan can be cancelled, and the reconstruction started.

MPEG reconstruction

If the raw recovery mode described above has been used, then the files will be a series of MPEG files. These can be double clicked and viewed Windows Media Player. However, they can not be written to a new DVD and played on a domestic video recorder. There are several options. There are many applications that can be used to create Video disks from MPEG files, and often such features are built into DVD burning programs such as Cyberlink Power Producer - 2 Gold. The CnW Rebuild video files (when complete) will allow mpegs to be merged, and a video disk image recreated.

-0-

Rebuild video disk files

[Home](#)

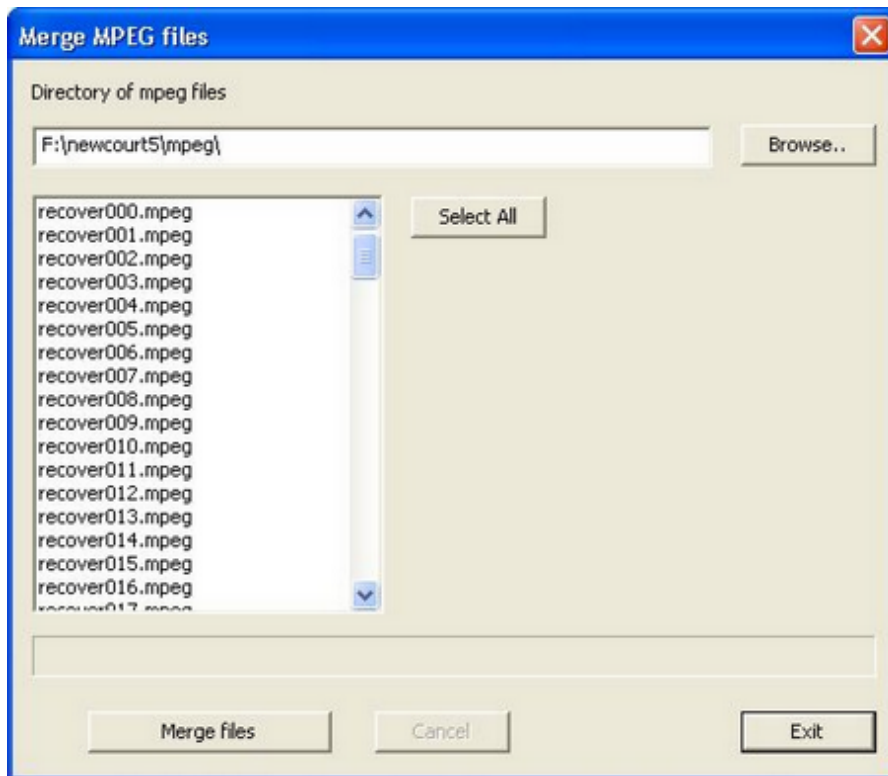
Some data recovery programs will extract MPEG files from damaged DVDs, but CnW goes one stage further and will recreate a complete video disk image, with no extra tools required.

This is a tool takes MPEG files, resequences them, and generate the relevant .IFO and BUP files.

Most data recovery programs will allow recovery of the basic MPEG and then require external software to convert these files into a format that will play on a standard DVD player. Sometimes Windows Media Player will display the recovered files, but at time it will only display the first few seconds.

The tool will allow for one or more mpeg files to be joined together, and relevant control files added. These files may then be copied to a DVD and run on a standard video player

The first stage of reconstructing the video structure is to use the function Create Video Disk from MPEGs which is found under the menu Tools. The program will create a new subdirecty VIDEO_TS and merge all the mpeg files into a series of VTS_01.1.VOB files, upto 1GB in size. There are then two control files, VIDEO_TS.IFO and VTS_01_1.IFO



The merge operation is very simple. Select the directory with the required MPEG files. The files must be in the correct order so it may be necessary to rename some files to ensure they are in numeric sequence. By default, CnW Recovery will create files with a 4 byte numeric extension, which will be sorted as required.

The first stage is to select the files to be merged. The select all button is the most common way to do this. When Merge Files is selected, all files are merged into a sequence of chapters. If more than 99 files, files are merged in groups to keep the total number down to 99. The files are then stored in a new directory, VIDEO_TS.

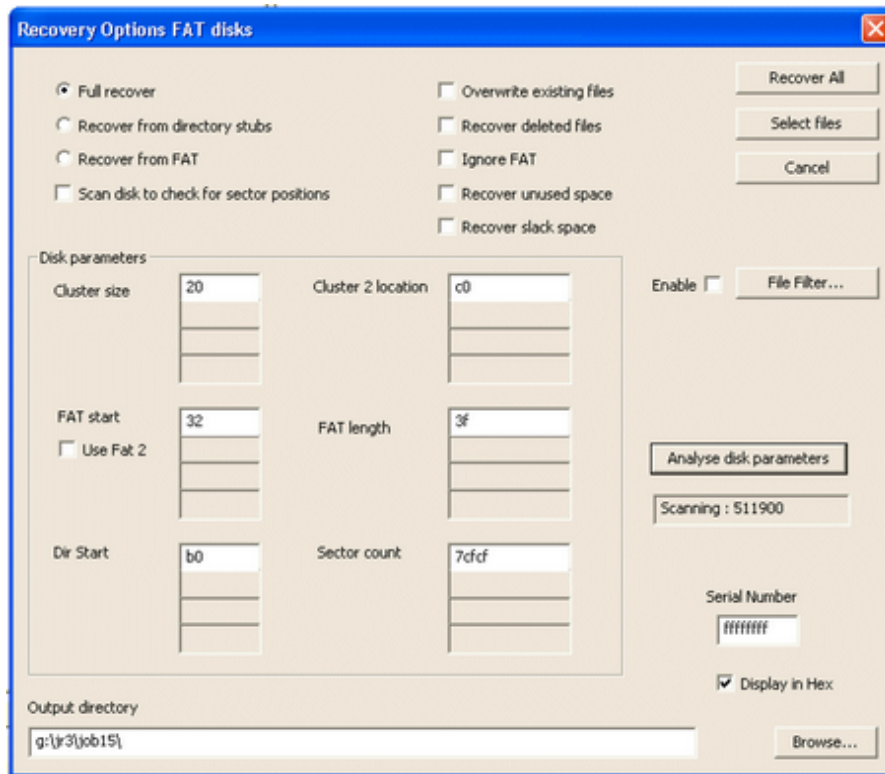
Creating DVD disk

Having created the DVD image it can be tested by double clicking on VIDEO_TS.IFO and typically Windows Media Player will display the file. To create a physical disk, the files must be copied to a DVD using one of many DVD writing or burning packages. Some software packages (such as Roxio) detect that the files are a video disk and will not let files be written as a data DVD. For these packages, it is necessary to use a DVD duplication routine. Examine the DVD writing software manual for more details

FAT Disk recovery

[Home](#)

This option is displayed for all FAT disks, ie FAT12, FAT16 and FAT32.



By double clicking on any of the parameter boxes the sector will be displayed.

There are three main restore modes possible for FAT disks, and then several options relating to these modes.

Full Recover.

This is probably the most common mode for restoration. The program will attempt to restore the files as in the normal operating system. However, it is very tolerant of corruption and will often restore many good files from the disk

Recover from directory stubs

This is a mode where the whole partition is searched for possible subdirectory stubs. This a sector that follows the pattern of a subdirectory, or even root directory. The program will then extract each file. Where possible, the program will try and determine the path of the file. If it is not possible, a new subdirectory 'dirstub0' will be created.

Recover from Fat

In this mode, the program will read the FAT, and restore files from the chain. Thus chains of files will be restored, but no attempt is made to extract filenames. This should be viewed as a fairly last resort measure, but can occasionally rescue files that would not otherwise be found.

Recovery options

Overwrite existing files

In this mode, output files will be automatically overwritten if they already exist. When this option is disabled, files will be renamed but adding an extension, eg .000, .001 to the name

Recover deleted files

This is a mode where files that have been detected as deleted will be restored. Although this option can often work very well, there are two potential problems. The first is that when a file is deleted, the space it occupies is made available for re-use. It is therefore possible that the original file is overwritten by a new one. The second problem is that when a file is deleted, the FAT is also cleared, so no details of a fragmented file is retained. If the file is not fragmented, then a good recovery can be made. See notes for [FAT32 deleted file recovery](#)

Ignore FAT

On some disks, or other media, the FAT is either corrupted, or deleted. One symptom of this problem is when files are restored, but truncated to 16K or 32K. When the ignore FAT option is enabled a new dummy FAT will be created. This assumes that all files are sequential, and on a lightly used disk, with short files, a very high success rate can be expected. With very long files, on a full disk, files may well be corrupted

Scan disk to check for sector locations

This is a very specialized option. On a disk that has been physically recovered, it is possible for sectors to be in the wrong location, ie areas of the disk have apparently moved. When this option is selected, the whole disk is scanned, and directory entries are detected. Each directory entry has a pointer to itself, and so it is possible to determine if the sector is in the correct location. A table is then built of possible sector offsets, and zones on the hard disk where these errors are found. It is not always possible to detect the exact boundary of the sector shifting, but on this type of error, this option does improve the restore rate considerably

Recover unused space

This is an extremely useful option, that will scan unused sections of the disk, and try and extract files. If it comes across a valid start of a file, it will produce files based on the signature found. A good example could be lost pictures, or jpegs

Recover Slack space - Forensic option

Slack space is the space that is at the end of a file, when the file length is not an integral length of a cluster. As an example, for a disk with a cluster size of 4K, when a file of 1K < 9K, or 201K is written there will be 3K of slack space at the end of that file. The data in the slack space

can be very varied, but could be the contents of memory when the file was written, or what was on the disk before, or a mixture. For forensic investigation, it can give very useful clues to what the user may have thought had been deleted from a system. For recovery purposes, it could add a bit more to a corrupted file.

Analyse disk parameters

FAT disks have a fairly typical range of values. For instance there can only be one or two FAT maps. A cluster size must always be a multiple of 2. If the program detects values that do not make sense it will suggest running the Analyse disk parameters function. This function scans the disk and looks for subdirectories. By finding at least two subdirectories it can work out many of the parameters required to recover the disk. These parameters will be loaded onto the screen, and can be edited if required. Occasionally some trial and error may be required, or direct examination of the proposed sectors.

Errors

A typical error is for files recovered to be truncated. Often they may be just 4K or 16K in length. The normal reason for this is that File Allocation Table (FAT) has become corrupted. Sometimes this will be detected when starting to read the disk with a message indicating that dubious sectors have been discovered. To recover from this type of error just tick the Ignore FAT box and run the recovery again. Most files on a FAT disk are sequential, and so a corrupted FAT can be guessed, but if the file is very long, or fragmented the file may be recovered at the correct length, but there may be corruption

FAT Disk parameters

The disk parameters are the values that FAT disks define themselves with. Thus any conforming FAT disk, or any capacity, may be read by setting these parameters up. CnW software allows for 4 partitions, and each partition will have its own set of parameters. The parameters can either be displayed in decimal or hex, depending on personal preference. A very useful feature of CnW software is that the default parameters can be overwritten for special recovery purposes. For instance, if you received a corrupted disk where the main directory was in the wrong location, the start directory parameter could be set to see the directory. As these values are not written to the drive, incorrect values will not damage the drive, but files may be recovered correctly.

The best way to initially set up the parameters if the automatically selected values are wrong, is to use the Analyse disk parameters function.

Parameter descriptions

Cluster size. A cluster on a FAT disk is the smallest number of sectors that can be allocated to a file. On a FAT 16 disk, there are only 65536 possible clusters, and so for any disk above 30MB, a cluster has to be greater than 1 sector. The chosen cluster size for a disk is a compromise between speed and wasted space. A cluster size of 8 means less house keeping, but a small file will always occupy the 8 sectors. To determine the cluster size it is necessary to look through the disk and find the start of small files. The gap between these files will often be the cluster size. A cluster size is always a power of 2, so only 1, 2, 4, 8, 16, 32 and 64 are valid values.

Cluster 2 location. Usable data area on a disk always starts at cluster 2. For a FAT12 and FAT16 disk this is the sector after the end of the root directory. All cluster locations are therefore based on this value. If for instance a disk recovered files, but the first sector of each file was incorrect, this could be caused by the Cluster 2 value being one out.

FAT Start and length. The large majority of FAT disks have 2 FATs (File allocation Table). They are always at the start of the disk, after any boot and loading programs. The two FATs are always sequential and so the length of the first FAT will enable the location of the second FAT to be calculated.

Directory start. For FAT12 and FAT16 disks, the directory start can be determined by finding the end of the second FAT. The length of the directory is then the space between the FAT end and Cluster 2. For FAT32, the root directory is located anywhere on the disk, and like subdirectories, it's allocation is determined by the FAT. It can therefore be fragmented, and of any length. For recovery, all that is required is the sector of the first location.

Sector count. This is not a critical value, but is used to try and prevent accessing areas outside of the partition. If in doubt, enter a larger value, rather than a smaller value.

It should be noted that the sector values are absolute on the disk, and not relative to the partition.

How to recovery FAT disk when boot sector and one FAT is missing

[Home](#)

A common problem with disks, or memory chips is when the start of the disk is overwritten. The following notes show how to recover such a disk.

Stage 1

The first stage is to identify the type of disk for the program to process. If there is no boot sector, when the Recover function is selected, the partition analysis screen will be displayed with a message requesting that analysis should be run. Once the analysis function is run, the operating system should be displayed on the screen - top left hand corner of the dialog box. For many disks, it is possible to cancel the analysis, and still get the correct operating system. For a FAT disk, it will display either FAT32 or FAT16

Stage 2

After the partition details have been set up FAT options menu will be displayed, typically with all figures set at zero. These values need to be filled in, and an element of trial and error may be required.

Cluster size and Cluster 2 location

The first function to use is the "Analyse disk parameters" function. This will try and determine the cluster size, and the location of Cluster 2. It does require there to be at least 2 sub directories on the disk, so there may be problems on the odd memory chip with no subdirectories. Fortunately, most cameras do store files in subdirectories, so this function will work.

The cluster size is the minimum number of sectors that are allocated to a file. With a large disk, there can be many hundreds of thousands of sectors which would be a large job for an older style computer to track. (FAT was developed with MS-DOS in about 1980). To make the job simpler, sectors are allocated in groups, or clusters. Typical values may be 4, 8, or 16. On FAT16 disk, there can only be 65536 clusters. So for a 1GB disk, you would need 32 sectors per cluster.

A very important value on a FAT disk is the location of cluster 2. This is normally calculated by allowing for the following bits of information

- Boot sector
- Reserved sectors
- Operating parameters system sector
- FAT map 1
- FAT map 2

Directory - on FAT12 and FAT16. FAT32 root directories can be located anywhere
Cluster 2

On a disk with missing boot sectors etc, this value may need to be entered by hand, but the analyse function will often calculate it for you.

FAT start and FAT length

It is possible to restore files without a FAT, but this will give problems with very large files, or when the files are fragmented. CnW Recovery will operate with just a single FAT (there are normally 2).

The FAT is always stored near the start of the disk and will normally start with the hex bytes F8 FF followed by numbers that typically increment by one. The numbers are 2 bytes long for FAT16, and 4 bytes long for FAT32. They are also little [endian](#). Sometimes the only way to work out the value for the FAT is to look through the start of the disk for nicely ordered numbers.

If the FAT start value is entered for what is actually the second FAT, the program will still work. If both FATs are known, but FAT 1 is corrupted, the check box for Use Fat 2 can be checked.

Directory start

The directory start is an important parameter. When ever possible, the directory start should be set to the start of the Root directory. Where this is not possible, if the program looks at any subdirectory start, it will then attempt to restore the complete tree from that node.

Directories are based on 32 byte entries, the first 11 bytes giving filename (in 8.3) format. The remaining bytes store location, file size, date etc. A directory may also contain other entries which are the long file name description. Details of this are beyond the scope of this documentation, so please look in [links](#) for pointers for further reading.

A subdirectory entry, is the same as a root directory, except the first 2 entries are always ".", "" and "..".

-0-

BIOS Parameter FDC descriptor for FAT

[Home](#)

The BPB is the first block of a FAT partition and describes all the critical details of how the disk is laid out. There are variables that are system dependent often based on size of disk.

```

000000  EB 3C 90 4D 53 44 4F 53 - 35 2E 30 00 02 01 02 00  è<•MSDOS5.0
000010  02 00 02 60 F4 F8 F3 00 - 3F 00 FF 00 20 00 00 00  `ðó ? ý
000020  00 00 00 00 00 00 29 B7 - E1 51 58 4E 4F 20 4E 41  )·áQXNO NA
000030  4D 45 20 20 20 20 46 41 - 54 31 36 20 20 20 33 C9  ME FAT16 3Ė
000040  8E D1 BC F0 7B 8E D9 B8 - 00 20 8E C0 FC BD 00 7C  ŽŃŁō{ŽŮ, ŽÄü½ |
000050  38 4E 24 7D 24 8B C1 99 - E8 3C 01 72 1C 83 EB 3A  8N$}Š<Á™è<rřë:

```

Details for FAT12 / FAT16

Bytes 0x0B-0x0C	00 02	0x200 or 512 bytes per sector
Byte 0x0D	01	1 sector per cluster. Possible values are 1,2,4,8,16,32,64
Bytes 0x0E-0x0F	02 00	2 reserved sectors FAT starts at end of reserved sectors.
Byte 0x10	02	Number of FATs - 2 is normal
Bytes 0x11-0x12	00 02	0x200 or 512 root entries to the directory
Bytes 0x13-0x14	60 F4	0xF460 number of sectors
Byte 0x15	F8	Media type F8 fixed disk FB removeable disk
Bytes 0x16-0x17	F3 00	0xF3 Sectors per FAT
Bytes 0x18-0x19	00 3F	0x3F Sectors per track - with modern disks this has no real meaning
Bytes 0x1A-0x1B	00 FF	0xFF Number of heads, as above of no real meaning any more
Bytes 0x1C-0x1F	20 00 00 00	0x20 Hidden sectors. This is the number of sectors from the physical start of the disk. ie, It should be the address of this sector
Bytes 0x20-0x23	00 00 00 00	The total number of sectors. If the value fits in Bytes 0x13-0x14 this field is blank If the number of sectors is greater than 16 bits,

this field is used.

Bytes 0x26-0x2A 29 B7 E1 51 0x51E1B729 Volume serial number

Bytes 0x2B-0x35 NO NAME Volume Name

Bytes 0x2c-0x3D FAT16 File system, such as FAT12, FAT32

Bytes 0x3e-45 Resevered for future use

Additions for FAT32

Bytes 0x24-0x27 Sectors per file allocation table

Bytes 0x2c-0x2F Cluster number for directory

These are the values that are used in the [Recover FAT](#) function

-0-

Missing directories and files on a FAT disk

[Home](#)

Recovery of files on a [FAT](#) disk does depend on good directory files existing. If a directory file is broken, or has become corrupted, the files in that directory - or subdirectories of that directory, will be lost.

With CnW Recovery it is possible to scan the disk for all directory trees, and then extract the files from them. The option to use is 'Recover from directory stubs'. In this mode, the program will scan from the start of the disk to the end of the partition looking for any sector that is the start of a subdirectory. It will then extract all files from it. Where possible, it will also try and determine the parent of the directory, but this is not always possible, and in these cases, a new subdirectory will be created within the 'dirstub' directory. Each new directory will have a name such as dir258 or dir4398

-0-

How to recover FAT disk when boot sector is missing

[Home](#)

The partition boot sector is used to define all parameters on a FAT disk. It will include cluster size, FAT lengths, and directory location, along with the directory length.

The media BIOS sector is normally the first sector on the partition. When this sector is missing, the details it normally contains must be filled in. A very good start is to use the Analyse Disk Parameters function. This function will scan the disk and calculate certain values based on the following.

Cluster Size

By finding two subdirectories, it is possible to work out the cluster size in sectors

FAT Start

A FAT normally starts with the hex codes F8 FF. A FAT is also always near the start of a disk, so only the first 1000 sectors are searched for a FAT

FAT Length

If both FATs can be found (in the first 2000 sectors of the disk) then the FAT length can be calculated

DIR Start

For a FAT12 or FAT16 disk, the directory starts just after the second FAT. For FAT32, the root directory can be placed anywhere.

Cluster 2 Location

Cluster 2 is the location that data storage starts. For a FAT12 and FAT16 disk, this is the location after the directory. For a FAT32 disk, is normally the location after the second FAT map and may also be the start of the root directory. Fortunately, it is possible to calculate this location from finding the location of any two subdirectories.

With much of data recovery, the automatic analysis may produce the correct results, but at times, they may need to be tinkered with .

-0-

FAT 32 deleted file recovery

[Home](#)

With a FAT32 disk that contains deleted files recovery is not always totally reliable. Never the less, CnW Recovery program does do much more analysis than many other software programs but below are described fundamental issues.

When a FAT disk file is deleted, two main things happen

- The file entry is marked as deleted, by setting the first character in the file name as a 0xE5
- The File Allocation Table is cleared

On Fat 32, the high order cluster pointer values are also cleared.

A FAT directory always uses a cluster number pointer to indicate where the file starts. For FAT 12 and FAT 16 this a 12 or 16 bit number, stored in two bytes at offset 0x1a and 0x1b in the directory. For FAT32, the pointer is 32 bits, with the extra two bytes (16 bits) stored at offset 0x14 and 0x15. It is these final two bytes which are (for some reason) also cleared when the file is deleted. Therefore with a FAT32 deleted file, only the lower 16 bits are available to determine where the file starts.

CnW Recovery software does not give up at this point, it will examine the file extension and for many common file type, it will therefore know how a file should start. For instance, a Zip file always starts with the characters PK. By knowing this, possible file starts can be examined, based on the lower 16 bits of the cluster number and there is a good chance that the required file can be found. However, without human intervention, this can not be 100% reliable, but it is quick, and automatic.

The second problem with any FAT recovery is that the file allocation table is also deleted. The initial approach is to assume that the file is sequential, and often this is correct, and so valid files are recovered. CnW are working on enhancements to this procedure which will increase the likely hood of only getting good files by only recovering files in clusters marked as unused. Some extra fragmented files will therefore be recovered intact.

Which recovery mode to use?

The FAT recovery screen has two useful recovery modes which may produce different results

- Full Recover
- Recover from directory stubs

For a disk that has just had some files deleted, the Full recover will work well. Deleted files will be recovered and written to the output directory, prefixed by !deleted

For a disk that has been used a lot since files have been deleted, the Recover from directory stubs is more likely to detect and recover all files. This is slightly more exhaustive than the Full recover, as it does not rely on an intact directory structure. Typically it will find files and subdirectories that can not be placed in a tree, and so files there will be many dirstub dummy directories created. The log function will indicate which files had been deleted by the 'D' in the flag column.

When the disk is being scanned, the display will indicate the number of Deleted FAT32 files that will be recovered. These are ones that the program has searched the hard disk for to locate the start of the file, of the correct type, in a known empty location.

-0-

FAT File allocation table validation and correction

[Home](#)

With a FAT disk, the location of every cluster in a file is determined by the file allocation table. There are in fact two such tables, and on a good disk, both tables will be identical. The most common type of failure is for either sectors to fail, or parts of the table to be overwritten. In these cases, the second table can be used.

An unusual failure is when sectors are partially corrupted, often by data bits failing, sometimes seen in memory sticks. This can lead to an apparently good table, but one that is not possible to use. The errors can cause the file to chain to incorrect sectors, or loop to a single cluster.

CnW Recovery software makes several tests on the FAT and will attempt to fix many possible errors. Types of errors that are detected are as below

- Duplicate cluster values
- Variations between FAT1 and FAT2
- Clusters that point to themselves
- Cluster strings that do not terminate

The most common (default) fix for these errors is to set the pointer to the next cluster. For a sequential file, this is the correct answer, but for a fragmented file, it is just possible that a fragment jump could be missed, though this would be a case of double bad luck.

For each change, an entry is made in the Forensic Report giving details of the modification.

-0-

Recover FAT32 disk when it has been reformatted as NTFS

[Home](#)

There are cases of a FAT32 disk being reformatted as NTFS. This means that much of the file structure is lost, and also the FAT can be overwritten or corrupted. Fortunately, as long as too much data is not written with the NTFS structure, many files can be recovered.

There are several stages in this operation, as described below

Start the program, and skip the wizard. It is now necessary to indicate that the disk is to be treated as a FAT32 disk, rather than an NTFS disk. This is done using the Partition function, and selecting the operating system as FAT32, rather than NTFS.

Now select the Recover function, and the FAT options screen should be displayed. There may be several error messages displayed as the FAT parameters will not be known. Select the function Analyse disk parameters. The analyse function may take some time as it is trying to find old FAT32 directory entries. Once found, it will populate the FAT parameters as for the original FAT disk.

To recover the FAT files, the best option is probably Recover from directory stubs, along with Ignore FAT. The program is now set up to recover the original FAT32 files.

Because the FAT is almost certainly corrupted, it has to be ignored, and so files that were originally fragmented will be corrupted. The success rate does depend on many variables, in particular how much data has been written with NTFS, which would overwrite old FAT32 files. The directory structure may also be rather limited, and there will be Lost_dir directories as the parent node may have been overwritten.

-0-

exFAT data recovery

[Home](#)

Microsoft has released a new version of FAT32 called exFAT, extended FAT. It is designed mainly for portable drives using computered with limited computing power, such as cameras and hand held devices. The main advantage is that it no longer has a 4GB file size limit and has better operation with disk drivers larger than 32GB. The format has many similarities to FAT, but the major difference is in file allocation. There is a bitmap used for cluster allocation. There is a FAT to handle fragmented files, but if a file is not fragmented, the FAT is not used. The benefit is that when writing to a disk, a FAT does not have to be updated with every cluster written, and the performance increase can be dramatic. It is now very rare to find a fragmented exFAT file, which means that recovery should be easier, even when much of the disk has been damaged.

32 GB is a limit that Microsoft has tried to implement on FAT32 disks for performance purposes, even though drive happily work with 1TB of data. ie, the limit is just the maximum size that Microsoft will format the drive to, though there are many free utilities to overcome this restriction

The CnW Recover routine will recover deleted files and also scan unallocated area for any lost files

-0-

Linux and Unix recovery

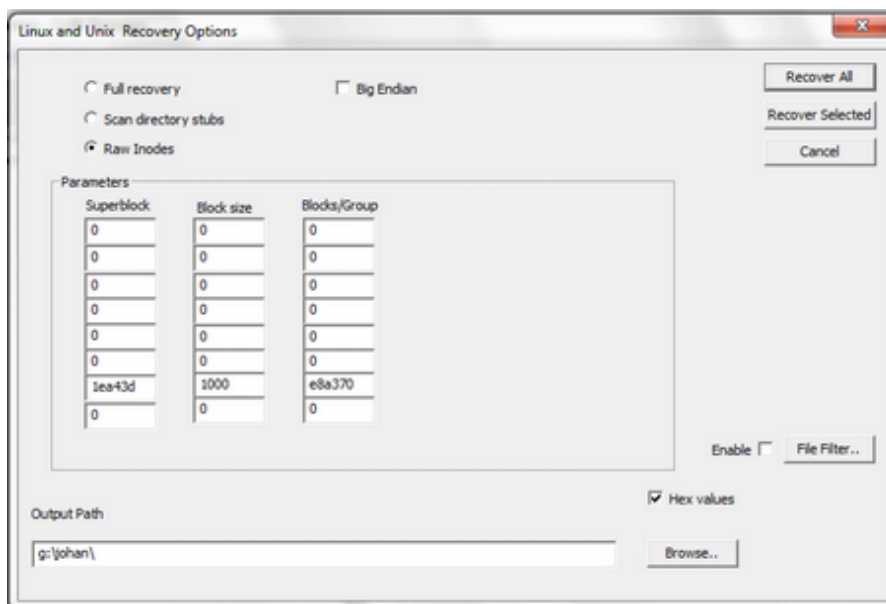
[Home](#)

Linux and Unix disks are not very common on their own, but are often part of a NAS (Network Attached Storage) system. This could be a single drive, or part of a RAID 0 or RAID 1 for small systems, and RAID 5 for larger, more secure systems

CnW Recovery will detect the following types of Unix

- Ext2/3
- Ext4
- ReiserFS
- XFS

When detected, the following screen will displayed



The most important sector on a Unix disk is the Superblock.

There are three basic modes to recover Linux disk by

- Full recovery uses the existing directory structure to discover all the files. If the directory structure is damaged, a full recovery will not be made
- Scan for directory stubs will search for each known inode and recover files this way. It is very possible that orphaned files will be found and these will then be stored in directories with a dummy file name
- Raw Inodes is (currently) for XFS and Reiser only. This scans the complete disk, block by block for possible iNodes. It then reconstructs where ever possible the file system, even when iNodes have been deleted and otherwise removed. The process can be slow, but it does result in files

being found that is otherwise impossible with most types of recovery software. Recent results with a test Reiser FS disk recovered about 80% of the original files that had been deleted. NB more files appear to have been recovered, but there many duplicates.

When the forensic option has been purchased useful detail is added to the forensic log. This includes expected numbers of iNodes, locations of groups etc.

XFS deleted file recovery

It is often stated that it is not possible to recover deleted files from XFS. This is largely true as unlike NTFS, there is no 'I have been deleted' flag. Instead the critical iNodes are partially blanks to make them look free, and the tables to state where the iNodes are, and if used are also cleared. The CnW approach is in five stages

- Scan the complete disk for all iNodes
- Regenerate the blanked iNodes to give file size, and file type
- Scan all iNodes to generate directory structure
- Recover all files, and check file signature
- Verify when possible, the file length

This process will recover files from very damaged XFS disks, and still retain file names, dates and very largely, the complete directory structure.

Reiser Disks

Most Reiser disks are part of the HP Media Vault system. They can appear as a RAID, or just a single disk. It is gathered that the system was often sold with a single drive, and then another drive could be added, normally as a JBOD configuration. The proposed RAID-0 option was never implemented. For RAID setup see the [RAID drives](#) section.

The disk may be read in three ways, Full recovery, scan and raw. With Full recovery, the first stage is an analysis of all the leaf iNodes to try and establish a directory structure. The Scan and Raw modes go to a lower level and do not try and read the disk based on the directory structure, though will try and reconstruct the directories.

A useful feature of the program is that it will still work even when the main Superblock header is missing. This header is normally at sector 0x80 of the partition, and is recognised by the string RelEr2ER at location 0x34 of the block.

Ext4 deleted file recovery

When an Ext4 file is deleted (and the rubbish bin cleared) the iNode is blanked out. This means there is no information on file size, date, or most

importantly file location. Put very simply, recovery of deleted files with file name is impossible. HOWEVER - with the raw mode it is sometimes possible to recover files with the correct size and extension and date, but still no name. The raw mode will scan the complete disk for old iNodes and make use of them.

The result of this scan can be varied - it detects all iNodes that are not part of the normal file system and so file may be found more than once. As names cannot be attached, the files are checked for signature and then saved in relevant directories. The file size and date are correct.

Recover All or Recover Selected

Not all configurations can operate with Recover selected. If Recover All is used and only; y certain files are required, the recommendation is to use the file filter to select files based maybe on name, file type or date..

It is intended to support recover Selected for all Full Recover modes of operation, but scanned modes will rely of the file filter.

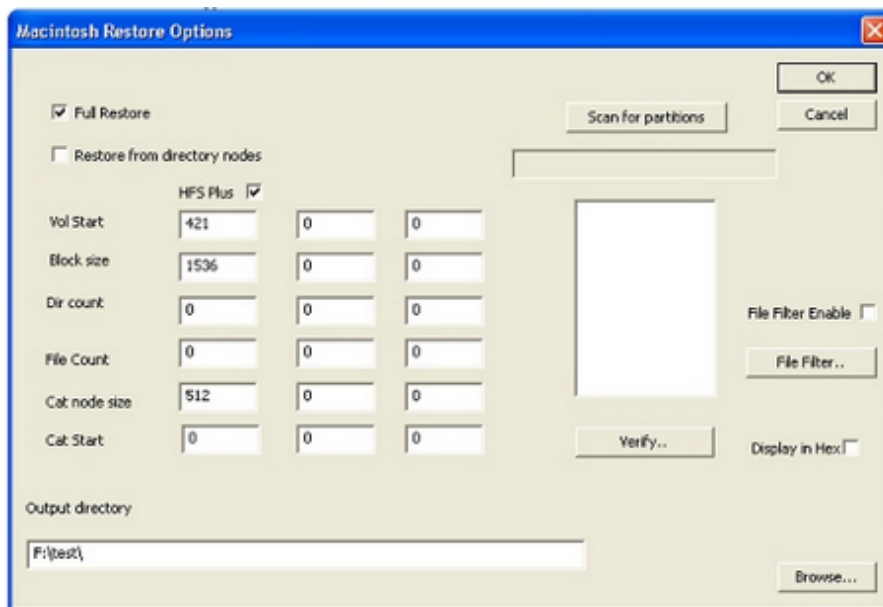
-0-

Macintosh Drive Data Recovery

[Home](#)

Macintoshes use a file system call HFS and HFS+. CnW Recovery, although it is a PC program has several tools and modes of operation to help recover files from disks that have become corrupted, or suffered partial failure.

When a Mac disk is selected for recovery, the following option screen is displayed - data is obtained from the [volume header](#)



The screen allows for three partitions to be recovered, and the basic entries for each partition are displayed and may be edited. The entries are as below

Vol Start

This is the first sector of a logical volume. For an HFS+ volume, logical sector 2 always starts H+. Sector 2 is the volume description block. There are occasions when the H+ sector is missing, or corrupted. In these cases the alternate backup copy is examined and used if appropriate

Block size

This is the logical block size that is used to allocate data, ie the amount used to write a 1 byte file. It is always a multiple of 512 bytes. For HFS disks, there can only be 64K such blocks on a disk.

Dir Count

This is the number of directory entries stored on the partition

File count

This is the number of files started on the partition

Cat node size

The catalog on a Mac disk is basically a file with fixed length records, or nodes. The size of the node is critical in recovering the disk. Typical

sizes for 80GB disks are 4096 or 8192 bytes (0x1000 or 0x2000). It is normally similar in size to the Block size, and will always be a multiple of 512 bytes

Cat Start

The cat start is the starting location of the catalog file

Recovery modes

There are three basic recovery modes that can be use.

Full Logical Recover will read the disk in a similar way to the operating system, but is very tolerant of any errors.

Recover from directory scan. This will read the catalog file and try and recover each entry in a leaf node. This mode will trap any file on a disk that has a broken directory tree. It is a very useful mode to use when there are failed sectors in the directory area of the disk. It is common on a damaged disk for files to be saved into 'dummy' directories with names such as dir_9865.

Recover from directory nodes will scan the complete disk for possible catalog entries. This mode would be used when there is considerable damage in the catalog area of the disk. The scan can be slow as it will examine the complete disk. However, it does not rely on a valid catalog tree extents information

Scan for Partitions

The scan for partitions function will scan the complete disk and try and determine if there are any possible starts to partitions. Any such starts are displayed in the box under the Scan for Partitions button. If the entry is double clicked, it will be placed into the column for the first partition, and the other parameters will be updated.

Scan for catalogs

The scan for catalogs will scan the complete disk and isolate the probable starts for a catalog. If the value is clicked, the entries in the first partition (the first column) will be initialised so that data from the selected catalog can be read.

Verify...

Processing Resource forks

On Apple disks, HFS+, files often contain both data and resource forks. There is also an important part of Metadata stored in the directory that indicates the application that should open the file. PCs tend to work just on file extensions, but Macs have a 'hidden parameter to assist, and so the file

name is not actually important. On OS X, there is a method of sharing this information on a standard PC disk. The method is called AppleDouble format. Put simply, for every file there is an associated file with the same name, but prefixed `._`. For example for `fred.doc` there will also be a file `._fred.doc`. This extra file will be at least 82 bytes long, and longer if there is an associated resource fork. By using these files, the Mac on OS X can treat a FAT disk in the same way it handles a native HFS+ disk, and no information is lost.

OS9 uses a different method - not yet implemented in CnW software.

The type of output required on a disk is chosen by the flag OS X on the options screen

A Macintosh will not read an NTFS disk, so if data is to be transferred to a MAC, a FAT32 must be used. Many external USB drives come as NTFS, and Windows will not reformat a drive as FAT32. It is therefore necessary to use an external program to format a drive as FAT32. One such program that has been tried is `fat32format.exe` that can be downloaded free of charge from the web. Writing to an external FAT32 drive is very slow unless the write cache is enabled. On grounds of performance, make sure the output drive properties are set for performance and not quick removal.

Shortcuts and Hypertext links

Many times when copying files recovered by CnW to a Macintosh, the Macintosh cancels with an error message of 'The operation cannot be completed because you do not have sufficient privileges for some of the items'. This has been found to be due to programs missing, or hypertext links missing. It has been found on both a network copy, or when the Mac is reading a FAT or NTFS disk locally. It is often found when the Macintosh being used to copy, is not the original system machine. To overcome this, CnW examines the resource fork and removes and `'slnk'` or `'hlnk'` on the resource fork. Files now copy without stopping. It does mean that some links will no longer work, but it will ensure that all data is copied.

-0-

MTF .BKF files

[Home](#)

A very common backup format is the native backup program within Microsoft NT, Windows 2000 and XP. It is typically used to write to tape, but the Microsoft Tape Format can also be used to create a single backup file, with the default extension of .BKF.

CnW Recovery software will recover these files, often as a two part procedure. The first stage is to recover the original .BKF file, and then open that file as if it was a disk image. ie use the Disk Image drive selection to select the backup file.

The recovery procedure will scan the backup file and display all files and directories. It should be noted that it is very common for all subdirectory names to be backed up, even when there are no files in the subdirectory.

The routine will work on backup files that are not complete which is a typical issue when the backup has been interrupted.

-0-

NTFS Recovery

[Home](#)

For NTFS there are several approaches that can be taken to restore files. There is no correct one to use, but they often have different uses depending on how the disk has been damaged, or what type data is being restored.

Notes at the bottom of this page give suggestions of modes to use for different types of failures. There is no practical limit on the size of an NTFS partition, and with an EFI disk header, it can be larger than 2TB.

CnW Recovery software will work with disks having the standard 512 byte sector size (0x200) as well as the 4K sector size (0x1000).

By double clicking on any of the parameter boxes the sector will be displayed.

Full Recovery

In this mode the program tries to restore the file in the same way as the standard operating system. It is very tolerant of errors, but if for instance the root directory structure is missing, the restoration may fail. In this case, use one of the options below,

Recover from file entries

This often the most useful mode for restoring files from corrupted disks.

It does assume a reasonably valid Master File Table (MFT) and it will read each entry in the table and try and restore the associated file. When selected, a second option will be displayed where the range of MFTs can be entered. This can be useful if a section of disk is causing the restoration process to hang or crash. In these cases, it would be possible to start and end the scan in sections.

An additional option is to Scan all MFT entries, when the whole disk is read testing for possible MFTs. If the Cancel button is pressed in this scan, the scan is stopped, but optionally it is possible to continue with the restore stage. This if it is known that all MFTs are in the 1,500,000 blocks, the scan can be canceled anytime after that, and restore will continue.

Recovery from MFTs is in two sections. First, known good MFTs are recovered, and save in the directory specified by the output path. The second scan is for MFTs that have otherwise been lost. These are stored in a subdirectory !recover_mft.

Select MFT Range

When restoring from MFTs, it is possible to select the range. If this option is not selected, the all potential MFTs are analysed, and files read.

Restore deleted files

NTFS marks a file as active or deleted, by using a flag in the MFT. When restoring the disk and selecting the deleted file option, the MFTs or directory is processed twice. The first pass, only good files are restored. The second pass, deleted files will be restored, but as known used sectors can be seen, the file can be marked as overwritten, and stored in a separate directory. Overwritten files may be good, but should be treated with caution as at least some of the file has been detected as overwritten.

Deleted files are stored in a directory !DELETED

Recover unused space

Recover unused space will do a raw scan of all sectors that have not been used. The data is saved in a directory call !recover_carving, and as in normal carving, will be in folders for each file type. On an NTFS disk, the carving will test for compressed NTFS sectors and process as required.

Recover Slack Space - Forensic option

Slack space on an NTFS volume is found in two areas. First, the space at the end of each file cluster, due to the fact that disk space is allocated in clusters, of say 2K length, but files are allocated space in bytes. A file of 13K, would therefore require 14K of disk space, leaving the final 1K as unknown data. This is slack space, and can be useful

within a forensic investigation. For data recovery applications, it is normally ignored.

Cluster slack space is stored in a file called Slack_clust.slk. Each fragment is enclosed by tags with the structure

```
<<clust:ssss-cccc>>.....</clust>>
```

where ssss is the first sector in the cluster, and cccc is the logical cluster number

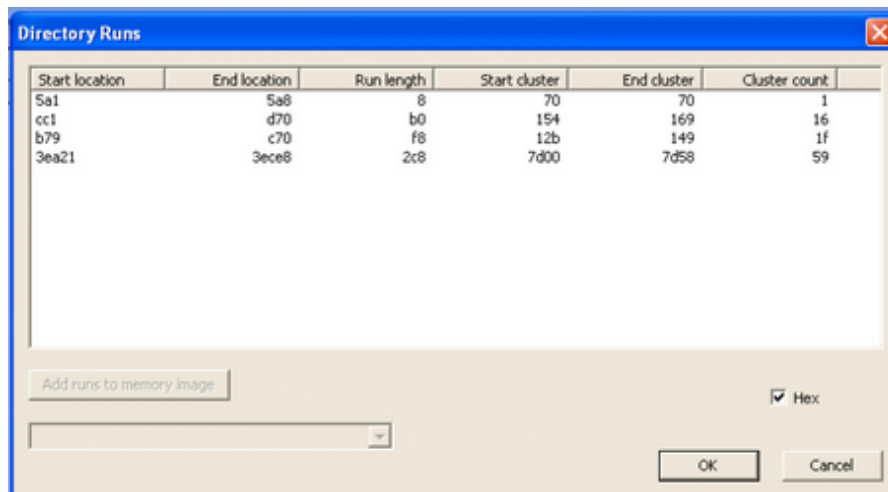
The second area of slack is at the end of each MFT. A short file, normally less than 3K can be stored in an MFT. This MFTs can contain more than just directory information. If the recover slack option is selected, all slack space from directories is stored in a file called Slack_Dir.slk, and placed in the output directory. Each entry is prefixed by the string

```
<<mft:mmmm-xxxxxx>>.....</mft>>
```

where mmmm is the MFT number and xxxxx is the sector number of the MFT. The data entry is terminated by <<\mft>>.

Display MFTs

On an NTFS disk the sectors for an MFT form part of a file. Typically, all the sectors are contiguous, but on a highly used, or full system, the file can be very fragmented. When Display MFTs is used, a list of starts and run lengths is displayed, as below



Start location	End location	Run length	Start cluster	End cluster	Cluster count
5a1	5a8	8	70	70	1
cc1	d70	b0	154	169	16
b79	c70	f8	12b	149	1f
3ea21	3ece8	2c8	7d00	7d58	59

Add runs to memory image

☒ Hex

OK Cancel

The start locations (absolution on the disk) and run length (in sectors) may be displayed in either decimal or hex.

When the input file is an image file, then it is possible, by using the Add runs to memory image, to scan the selected hard drive and add the relevant sectors to the disk image.

Analyse disk...

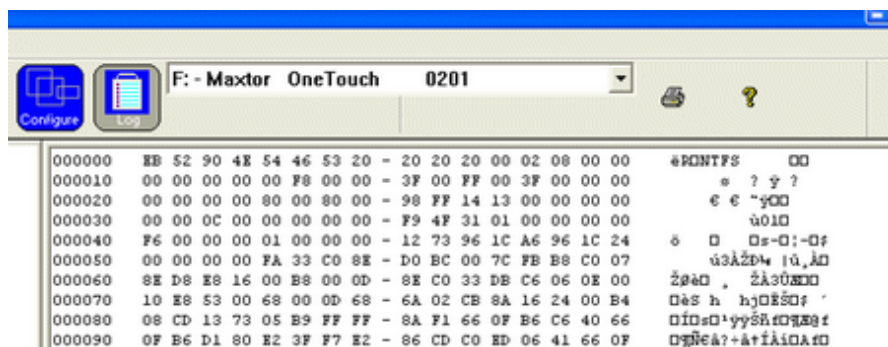
This is a function to assist in locating MFTs, and their size. For full details, see [Search for MFTs](#)

Disk parameters

There are 6 parameters, for upto the total of 8 partitions. It is these values that determine how a disk is read logically. With a working drive, these will be filled in automatically, and will not need changing. However, for a failed drive, they may need to be configured, or adjusted. File can often be recovered from a disk that failed during partition resizing by setting these values to one of the logical partition sizes for the disk.

Scan start

This is the start of the logical partition. A typical sector image is shown below, with NTFS in bytes 4 to 7



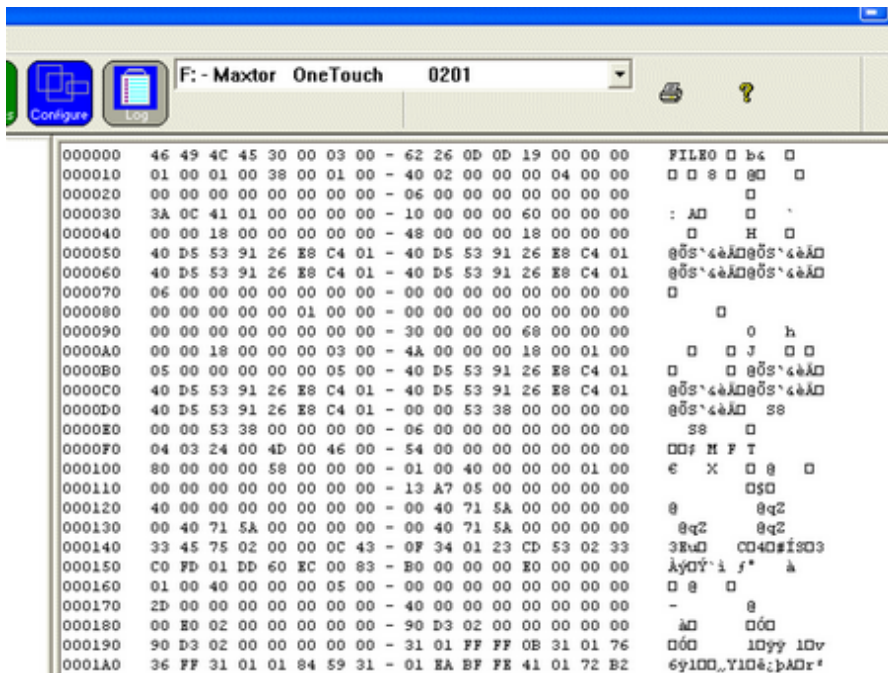
This value is critical, and for a single partition disk is often 63 (0x3f)

End Scan

This is sector location of the end of the partition. The value is not critical, and so if not known can be set to that of the size of the disk, or slightly larger.

MFT cluster start

The cluster start is the cluster number within the partition for the first MFT entry. Shown below is a typical first MFT



The location is worked out from the Start Scan entry, and is typically 0xc0000. This is the value in bytes 0x30 - 0x33 of the Start scan sector, saves as little endian, hence 00 00 0C 00

An MFT entry always starts with the string FILE0 or FILE* - the difference is due to two versions of NTFS. The root MFT has the string \$MFT within the sector as this is the (hidden) file name for the MFT file, ie the main NTFS directory details. An MFT entry is always 1024 bytes long, so 2 sectors in length. So all MFTs will either start on an odd or even sector number.

MFT Start sector

The start sector is the physical sector the first MFT is stored on. This is calculated by the cluster start * cluster size + Start Scan. For a typical single partition drive, it is 0x60003f. The Analyse Disk function will help determine the value for this entry, and the cluster number.

MFT entries

This is the expected number of MFTs. Most files and directories require a single MFT, though some files with long file names, or very fragmented, require multiple entries. The value in this field is not too important. If in doubt it should be set to a value too large. A value of 250000 will allow for over 200,000 files and could be a good starting value. If the value is 0, then set it to a suitable value as described earlier.

Cluster size

This is an extremely critical value. It must be a multiple of 2, eg 2, 4, 8 and for most disks above a few GBs in size, the value is 8

Alternate Data Streams (ADS)

The very large majority of PC users will never be aware of alternate data streams. They are a hidden part of a file that will not be seen with any standard DOS or windows tool. However, they are part of a file, and normally stripped off on recovery. However, with the correct tools these files can be used to hide data on a drive, and so CnW Recovery will extract these data forks.

CnW will produce a file for each data stream. For alternate streams, the file name will be appended with the string `-#-xxxxx` where `xxxxx` is the stream name.

How to recover after different modes of failure

[When operating system has been reloaded, and all data files lost](#)

-0-

BIOS Parameter FDC descriptor for NTFS

[Home](#)

The BPB FDC descriptor defines all the parameters required to logically read an NTFS partition. It is stored at a location pointed to be the Partition Table record, and typically it is sector 0x3F (63).

The FDC starts at byte 0xb, and although similar to a FAT FDC has differences. The example below is described in detail.

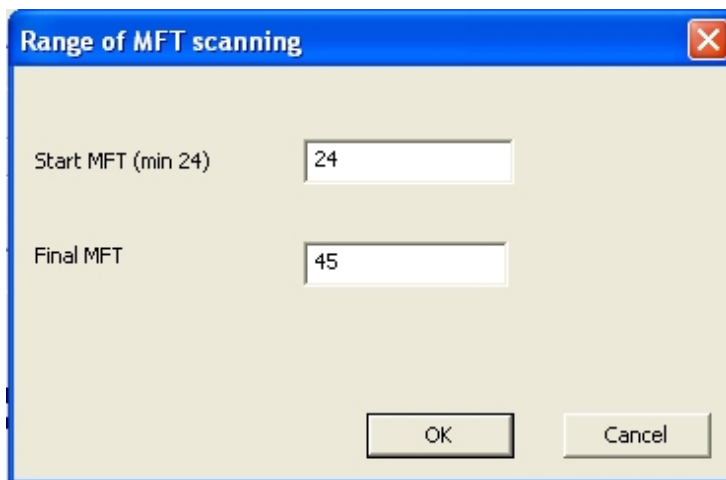
000000	EB 52 90 4E 54 46 53 20 - 20 20 20 00 02 08 00 00	ëR•NTFS
000010	00 00 00 00 00 00 F8 00 00 - 3F 00 FF 00 3F 00 00 00	ø ? ý ?
000020	00 00 00 00 80 00 80 00 - 2F 37 38 1D 00 00 00 00	€ € /78
000030	00 00 0C 00 00 00 00 00 - 73 83 D3 01 00 00 00 00	sfó
000040	F6 00 00 00 01 00 00 00 - A0 74 26 B4 90 26 B4 54	ö t&'•&'T
000050	00 00 00 00 FA 33 C0 8E - D0 BC 00 7C FB B8 C0 07	ú3ĂŽĐ¼ û,À

-O-

NTFS MTF range

[Home](#)

When restoring from file entries, the program will scan the range of Master File Tables (MFT) that it has determined. Sometimes, this may cause a problem, if a particular entry is very corrupted it may cause the program to loop, or crash. This type of problem should be reported to CnW Recovery, but in the man time, it may be possible to restore parts of the disk by setting the MFT range to be less than the full range. Thus, the user decide to restore just the first 1000 entries, or the entries in the range 4,500 to 6,000 rather than the complete disk.



-0-

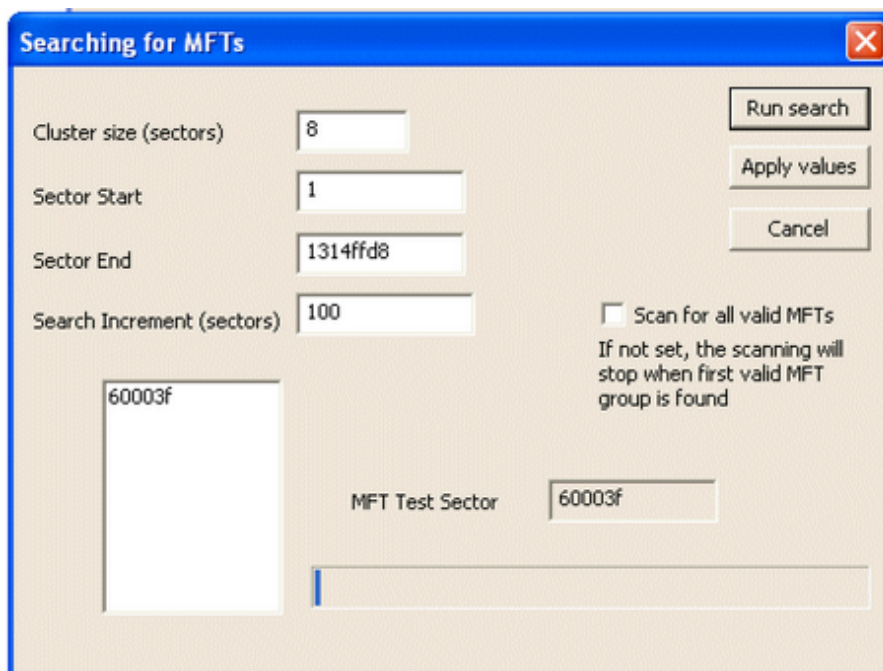
Search for MFTs

[Home](#)

Some disks are extremely slow to access. In this case searching for MFTs could take years rather than hours. The search for MFTs does a faster search looking every possible 512 locations for a possible MFT entry. This does assume that a run of MFTs will be detected, but it is possible that a run could be missed. It is therefore not a function that should be used if a complete forensic report could be required in court.

By finding MFTs, this is also the basis of a simple analysis of the disk structure, as it will also determine start of partition, and cluster size.

The values are displayed in Decimal or Hex, depending on the value set in the main NTFS recover screen.



The above information is filled in automatically by the program, there is flexibility to change it. In particular, the starting location of the search is determined by Sector Start. This feature can be used to start searching on an area of the disk known to be, for instance, the second partition

Run Search

The search increment is the number of sectors to jump between searches, so the display above would relate to every 256 (0x100) possible MFTs, for a normal 512 byte disk. (Some optical disks are 1024 bytes). A large search increment may skip over a section of the disk containing MFTs, a

very small increment will take a long time to search. When the disk is searched, two sectors are actually read, so that it does not matter if the start is odd or even. Once a block of MFTs are found, the program then searches backwards to the first MFT in the range. The status list box shows the start of any MFT run that has the first entry of \$MFT

Start sector and End Sector

These values define the range of searching. Typically use the default values, unless one knows where to look, and one wants to save time.

Scan for all valid MFTs

The \$MFT is a file, made up of MFT records. Each record starts with the string FILE0 or FILE*, and is 1024 bytes long. The first entry will always have a filename of \$MFT. The function scan for all valid MFTs will scan the complete disk for any run of MFTs that starts with \$MFT. For many applications, only the first one is required, and for a single partition disk, there should only be one.

For a disk that has been repartitioned, a full scan may well point to where old MFT runs have been found.

Apply Values

If the scan has brought up possible MFT runs, these may be applied to the main recovery program. If more than run has been detected, it is necessary to select the one required. It will then configure these values into Partition 0 of the disk. Thus if there are multiple partitions, it will be necessary to run this routine several times for each partition.

Cancel

The cancel button has two modes of operation. If the program is scanning, it will cancel the scan. If the scan has finished (or been cancelled) then this function will exit, and not update the main parameters

-0-

Files lost when NTFS reloaded

[Home](#)

It is a fairly common problem that many PCs these days are shipped with a recovery mode for the operating system that will reload a completely clean copy of the operating system, and not retain the files. Fortunately, this reloading does not normally include a re-format, so the files do still exist, but cannot be accessed.

To recover the files, one way is to track down all the old MFT entries. The new installation of the operating system will create a new MFT file, ie the index to all files, and the new MFT will be fairly short, ie just long enough for the files currently on the disk, maybe 10,000 files. To recover the older files, it is necessary to find the original MFTs. As all indexing of the locations of the MFTs is probably lost, the only way to work is to scan the whole drive. For this reason, on the [NTFS options](#) screen, one needs to select both the radio button and check box

- From File entries
- Scan all MFT entries.

When the Recover All, or Select Files button is pressed, the whole drive will be scanned. If a scan of the whole drive is not required, the End Scan value can be reduced from it's default maximum value. In the same way, the start

value could be increased. This setting of start and end is also useful if necessary to skip some of the media due to complete, or excessive sector failures. When scanning, the display will indicate progress, as well as the number of MFTs and Boot sectors found. If Cancel is pressed on the scan, the scan for MFTs will be cancelled, and the process will continue with file recovery, based on the number of MFTs detected to the point the cancel button was pressed.

File recovery mode works in two passes. The first pass will recover all known good files, ie ones from the current NTFS disk. The second pass will then recover all files that have been located from the scanned MFTs, as well as deleted files. The reason for this dual pass is so that file can be detected that could have been overwritten, or partially overwritten.

Files that are recovered, as part of the second pass are stored in a directory !recover

Files that are thought to be partially overwritten, are stored in a directory overwrite, or overwrite\!recover. Overwritten files are often corrupt, but can always be tried, as they contain the information / file data required.

Files recovered, but not valid, or subdirectories wrong

The most likely explanation for this is that the drive partitions are incorrect. Locations of files, as stored in an MFT is relative to the partition that the MFT is in. The starting point to fix this will be to run the partitions program and search for previous partitions

-0-

Cannot read first mft, copy failed

[Home](#)

The first MFT is a critical sector. There are actually two such sectors, and both are searched for before this error message is displayed. The way to recover from this problem is as below

In Recover function, the following options must be selected, enabled

- From File entries
- Scan all MFT entries

The program will then scan the whole disk for MFTs. It then does two stages of recovery which may look a bit odd if the first MFT is missing. The first stage is to try and recover all files that exist in the full MFT file - which in this case may not exist as a file. The second stage is then to recover all files relating to an MFT. Problems that can exist with this mode is that occasionally directory paths can not be resolved fully, so files may appear in invalid, or incorrect directories. CnW are working on this problem.

-0-

NTFS with confused partitions

[Home](#)

When a partition modifying program fails, an NTFS disk may be left in what may be best described as a confused state. ie it may be possible to find where the MFT file is, but it does not tie with the files. The reason for this, is a repartition program may move the location of the MFTs, and sometimes move the locations of programs. If this process fails in the middle, there may in effect be two groups of MFT entries, pointing to two groups of files. To recover the files, it may be necessary to recreate the partition information for the initial partition settings, and then another set of parameters for the second partition settings, and MFT locations. The disk is then recovered in two stages

This section gives guidance on to recover such a disk.

The first stage always is to establish where the actual MFTs are located. For this there are two tools within CnW Recovery to assist.

- The first tool is the [Partition analysis](#) routine. If partitions have valid headers, this function will assist in searching for partition starts, and hence pointers to MFT files, ie the disk directory information.
- If the partitions do not have valid headers, then it will be necessary to set the partition to be NTFS in the partition analysis section, and go to the NTFS Recover menu and use the Analyse disk function. This will [search](#) through the disk to find the start of MFTs.

At this point we may have the start sector of the MFT, ie a sector that starts with FILE and part of the way through has the string \$ M F T. This is the value that has to be entered in MFT Start sector. However, on recovery one may get lots of files, and valid filenames, but not valid files. This problem is due to the start of the partition being wrong. To establish the start of the partition can be time consuming, but very satisfying when you get the correct result. The start of the partition is determined in 2 stages. First run the recovery program and get file names and sizes. Secondly, run a raw recovery of an area of the disk to obtain many files that have known sizes and extensions. Typically a jpeg file is very good for this. Then one can match a jpeg file with a know size between the directory determined location, and the raw recovery location. A bit of simple maths with the indicate how the value for the start of the partition should be altered. This sounds complex, but is not actually too bad, it just takes careful thinking. As the master disk is never changed, and data is recovered onto a different drive, multiple attempts will not corrupt your data any further.

CnW is working on ways to automate this process.

-0-

hello.txt	40 bytes long
hello.txt-#-hidden	17 bytes
hello.txt-#-hidden2	26 bytes

-0-

Recovering when a new /different operating has been loaded

[Home](#)

Some times a disk is lost because a new operating system is loaded. This is normally accidental, but malicious cases are also known. The major problem with data recovery is that the disk is probably working OK, and looks OK, but has not got any relevant user data.

The procedure to overcome this situation involves reconfiguring the partition table to look for the older type of operating system, and probably reconfigure the media partition data parameters. This is not as complex as it sounds, and with CnW Software, each attempt can be tried and tested, so some element of trial and error can be used.

If a case such as this is considered possible, it is more important than ever that the disk is worked on as a data disk, and not try to run any programs on the disk.

An example could be if somebody has installed Linux over an NTFS partition. If the partition data is displayed, then a valid Linux disk will be seen, and no sign of the existing NTFS disk. As long as the Linux has not overwritten actual files, there is a good chance of data recovery.

There are two basic stages to be performed

- Set the partition to correct operating system
- Set the operating system details

Set partition for operating system

From the main data screen, select Partitions. For the purpose of this example we will assume that the original disk just had a single partition with NTFS. This is a very typical configuration, but certain manufacturers, such as Dell actually partition the main drive into 3 partitions to allow for certain data recovery procedures. (This means that the main data partition does not start at the start of the disk.)

For the first partition, select the operating system to be NTFS, and the relative sector should be 63 or 0x3f. The value 63 is true in over 95% of cases, for the first partition. The total number of sectors should be taken from the highest total number of sectors displayed in the list. The value is not too critical, but setting it too high will slow down possible recovery, and too low may miss some files.

The Cyl, head, and sect values are only displayed, and not actually

used. They need not be set to any particular value.

It should be noted that these new values will be remembered by the program, even though they are not written to the disk

Operating system details

When Recover is selected, the [NTFS Recovery](#) screen is displayed. Often in cases with the operating system overwritten, there will be no meaningful information on this screen, and it will be necessary to locate the MFTs and media partition sector. One bit of useful information is that many NTFS disks use the same basic parameters and the following parameters can be tried - the values are in Hex

- | | | |
|---------------------|----------|--------------------------------------|
| ▪ Start Scan | 0 | |
| ▪ End Scan | | Size of drive, a fairly large number |
| ▪ MFT Start Cluster | 0xc0000 | |
| ▪ MFT Start Sector | 0x60003f | |
| ▪ MFT entries | 0x20000 | |
| ▪ Cluster size | 0x8 | |

There is a function button, [Search for MFTs](#). This will scan the disk for first run of MFTs. It will try and verify that the MFT is part of a main directory, and not as often happens, just an MFT sector that has been moved somewhere. This value can then be entered into the main screen.

-0-

Deleted Partition

[Home](#)

It is a common error to accidentally delete a partition, and in doing so, all files are lost. CnW software will help you recover your files.

There are few stages, and if no data, or operating system has been added, a complete recovery will be possible.

The first stage is to determine the original partition, and full details are in the [Partitions Analysis and recovery section](#). If this successfully recreates the master boot record, then a normal recovery function can be done on the disk, for both NTFS and FAT.

If it is not possible to detect the partition information, then it is necessary to force the partition type to the one required. This is normally NTFS, but it should be noted that several large PC manufacturers do include small FAT partitions at the start, and sometimes the end of a disk. The user may think the disk is NTFS, but there can also be FAT partitions.

The relative sector, and total sectors need to be filled in with 'reasonable' values, along with the operating system. When Recover is then selected, each recovery function has an analyse mode that can be used to determine the partition values.

-0-

How to find and recover lost files

[Home](#)

Often with a hard drive a file or folder may go missing. This could be operator error, or a sector has failed within a directory tree. The following tools and procedures may assist in recovering such a file.

If a disk is fairly new the Wizard gives an indication of the number of sectors of a disk that have been used as a percentage of the whole disk. Thus if the wizard says a 100GB disk is 80% full and only 20GB of files can be found, it is possible that there 60GB lost or hiding. However, an, or well used disk will slowly write on all sectors of the disk, so this figure should be used with caution.

Recovery is dependant on the media and operating system used so the instructions below are system dependant.

FAT lost file recovery

On the FAT recovery menu there are a few very useful options. For deleted files there is an option button to recover deleted files. This will copy the files that have been deleted. Often the full filename will be restored, but on short file names, the file name may start with a '!' as the first character of the file name is use to indicate that it has been deleted.

When files or directories are missing, the 'Recover from Directory Stubs' can be very useful. In this mode the whole disk is scanned for subdirectories. It will therefore pick up directories that have otherwise been disconnected from the main directory structure. This can be caused by a sector failure or a glitch when updating directories. As this procedure is a brute force method, there may be cases when totally irrelevant directory stubs are detected, or the same one found more than once.

NTFS lost file recovery

NTFS is well structured to recover any file that has otherwise been lost. Each file has an entry in the MFT (Master File Table) and so by searching the disk for all MFT entries, most files can then be found. Each entry in the MFT starts with the letters FILE0 or FILE*, and there are also sumchecks to add to the search criteria.

On normal reading of a disk, the MFT is navigated using Index files, and if one of these is corrupted or damaged parts of the directory tree will not be found. By using the recovery option 'From directory stubs' the disk will be searched for MFT nodes. There are two ways this can be done, either by

logically reading the \$MFT file which is quick, or for fuller recovery, the whole disk is scanned.

When the 'From Directory Stubs' option is used it is common for a file to be found without a parent directory path. In this case a dummy subdirectory is created, all associated files are stored in a unique directory.

As with FAT, the NTFS recover option menu has the Recover Deleted Files button. When recovering files, these will be placed in a main subdirectory called 'DELETED'

-0-

Recovery from a drive with many bad sectors

[Home](#)

CnW Recovery software will recover from disks that do have a high number of bad, or failed sectors. The best way to proceed is to create an image file, probably in stages.

The following stages should be followed.

The disk image should be started, and if possible the complete disk imaged.

At various points the imaging may stop, but if it stops for more than 5 or 10 minutes, it is probably preferable to start skipping sectors. This can be done using the Configure/ [Hardware configure](#) options, and probably set the drive up so that if 10 read errors occur, then skip 100 sectors. This will allow skipping in sections. If the system keeps pausing for long periods, try increasing the skip value, to maybe 1000, or 10,000 sectors. If this keeps hanging, cancel and try the next procedure.

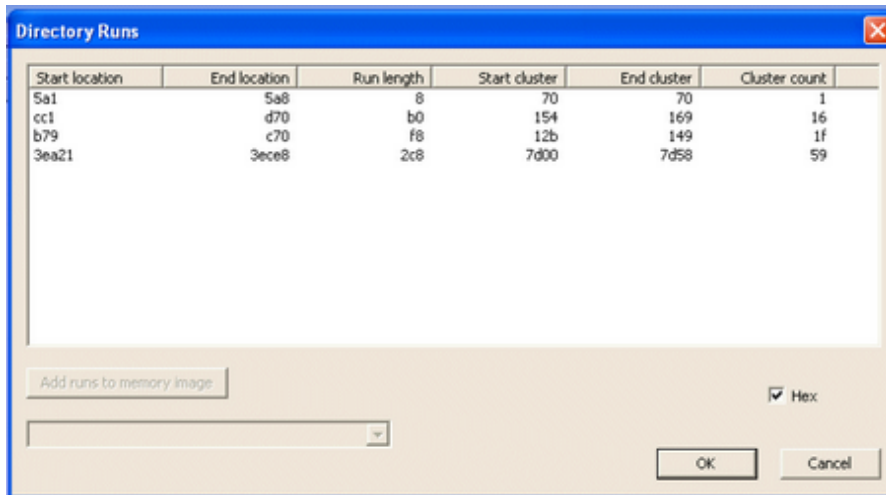
The directory image should be constructed

To construct the directory image it is necessary for the disk image to have enough information to determine the basic disk structure. For all disks this normally means a valid sector 0, (Boot sector), or one created by the [Partition](#) section of CnW. It is also necessary to have a the start of the partition imaged, and the start of the directory or catalog. The following values are only typical values for single partition systems, but are often correct

NTFS disk - partition start, sector 63 (0x3f).
MFT start, 6,291,519 (0x60003f)

MAC HFS+ - volume start 262,208 (0x40040)

If the above sectors are part of the image file then when Recover is selected, there is a button on the screen for View MFT or View Cat. When selected, it will display the location of all directory starts and lengths.



At this point, the failed hard drive should be selected, and the function 'Add runs to disk image' will now access the hard drive and update the disk image file.

After this stage there will be a disk image with the basic sectors required to navigate the files on the disk.

Final stage

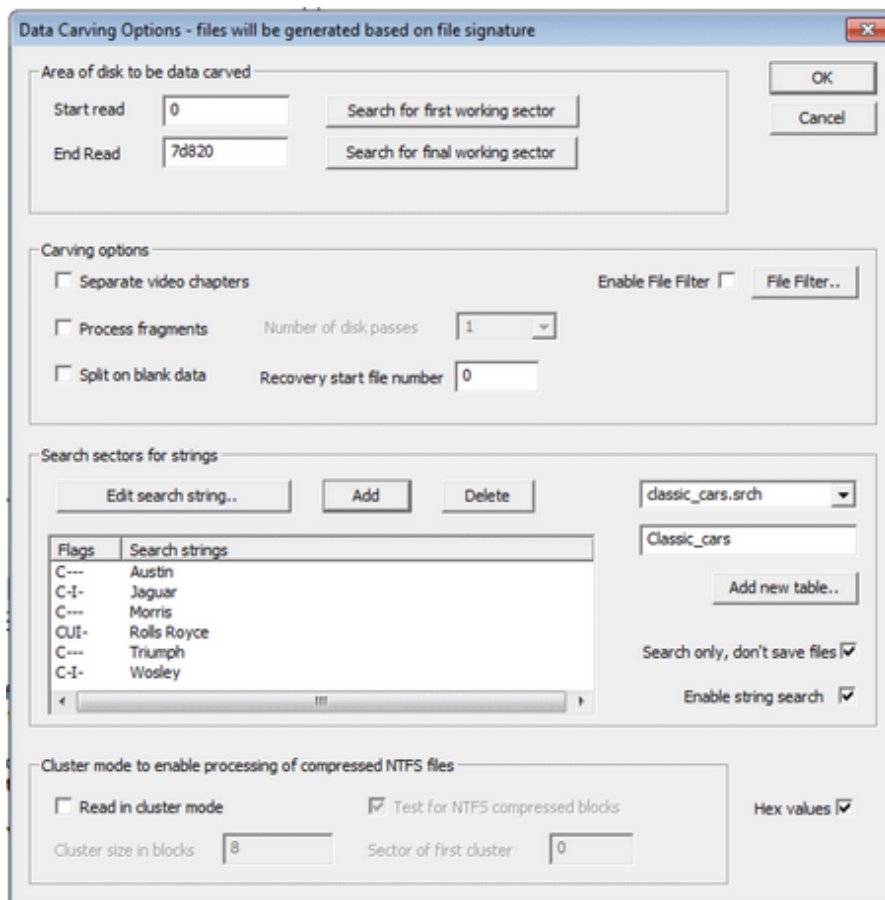
If a recovery is required, it will now be possible to do a 'dummy' recovery using the image file.

-0-

Data carving options options

[Home](#)

Data carving is processing data based on file content rather than using a file system. The disk, or the area selected will be scanned and when a possible file start is found, and new file will be generated, and placed in a subdirectory based on file extension. When possible, the file will be analysed further to generate a meaningful file name, or file date.



There are 4 sections to the data carving process

- Area to carve
- Type of carving
- Optional string search (Forensic only)
- Processing NTFS compressed disks

Area to carve

The carving process can either carve the complete disk (by default) or just select a specific area. One reason to limit the search could be if the final area of a disk is known to be blank. It can also be used to just carve a

particular partition. The sectors numbers are entered (in hex or decimal). The search for first or final working sector is typically used for CDs or DVDs to establish the are of the disk that can be read on unfinalised disks

Carving options

Separate Video Chapters

This mode is used to process video disks - in particular mini dvds. When it finds an MPEG file, it will then determine if a new chapter has been started, and then start a new MPEG file. Without this option, a DVD could end up producing just a single MPEG and this makes navigation (next chapter, etc) difficult.

Process fragments

This is a very power option when dealing with JPEGs and AVI files from a disk that has been fragmented. At the end of the original disk scan a list of possible fragmented files is displayed. At this point they can be selected for processing, and hopefully reconstruction the fragments found.

Split on blank disk

This will treat blank sectors, ie those filled entirely with zeros as the end of a file. Some files do have data that is blanks, so this option should be used with caution.

Recovery start file number

If it is necessary to restart the data carving process, by default the file naming will start recover0000.xxx. By setting the recovery start number to a higher value, the file naminmg can be set to start for instance at 10000, rather than 0. This means that multiple carving runs can save all the files in the same directory area, without a possible naming conflict. The number is always decimal.

Skip verify

An important feature of CnW data carving is that it verifies files, and with common file types it will try and create a more meaningful file name, or add the date etc. Very occasionally this can go wrong and maybe cause the software to crash. To avoid this, the verification can be disabled. This automatically also locks out any possible file defragmentation. When ever possible, files should be verified.

File filter

The [file filter](#) option can be used to select (or skip) certain catagories of files

Cluster modes

When the cluster mode is enabled, the program will only look for possible file starts at the start a logical cluster. When there are 8 sectors to a cluster this means that it will only look every 8 sectors, and

this will help reduce the number of false file starts. The program will automatically set the location and size of the clusters, but these values can be overridden. For NTFS disks that have been compressed, the test of NTFS compressed clusters will test each cluster to see if compressed. If it has been compressed, the program will read 16 clusters and try and decompress the data. On a non fragmented disk, the results will be good, but on a heavily fragmented disk, the results may be very variable. For more details on clusters see [Disk Clusters](#)

Search String

The [search string](#) option will search for entered strings when scanning the disk. There is an option do just a search, and not save any files at the same time. This is a forensic log option.

Multiple sets of search strings can be saved on the system in separate tables. To create a new table, enter a name in the box above 'Add new table.'. At that point a new table will be created and [strings](#) can be added. There is no limit on the number of strings, but the speed of searching is influenced by the length of the shortest string being searched for. The longer the string, the quicker the search.

-0-

Raw files

[Home](#)

The following formats are ones that are detected when running in Raw recovery, or image mode. The files are largely detected by the signature at the start, and then on some files, there is also further processing.

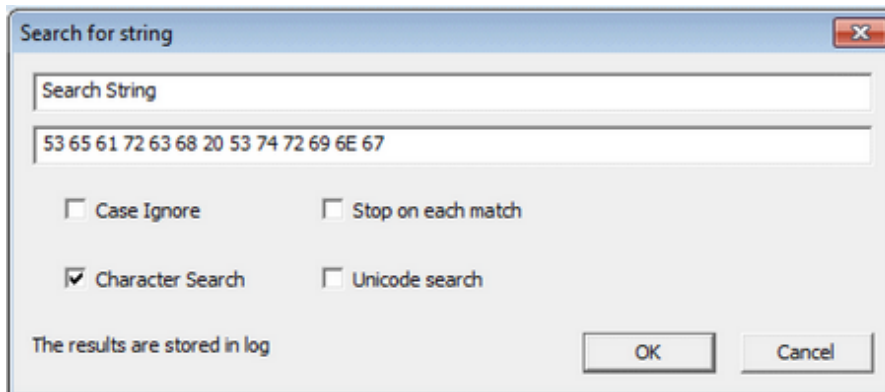
The list below grows on a regular basis, and if any extra formats are required, please e-mail a sample file to us at info@cnwrecovery.com and if possible it will be added to the system.

File extension	Type of file	Notes
abc	Flow chart data base	
ai	Adobe Illustrator	
aiff	Audio file	Length is determined from header
ani	Animated pointer file	
atn	ATN files	
avi	Audio visual files, movies	
bmp	BMP bitmap files	The length of the file recovered is determined by the header
cab	CAB	Compressed files as distributed by Microsoft
DOC	Microsoft Word document	There are many other documents that use DOC as an extension
exe	Microsoft executable file	Many files, such as DLL OCX have the same signature
jpg	JPEG image file	The file is parsed and length determined if the file is valid
mov	Movie file	For some version of MOV files the length is determined from the header
TIFF	Image files	The file length is corrected from data in header
ZIP	PK-ZIP files	Will be verified

-0-

[Home](#)

A very useful tool is the ability to search the complete disk for a possible string(s). This option is only enabled with the forensic package. The search can be set up to work with multiple strings, and mixed types of string. ie a string may be either straight characters, uni-code, or both searched for.



The disk will be searched at the same time as data carving takes place, if required, just a simple search and no saving of data.

Every sector / cluster is searched for the string, which can be any combination of characters, not just printing characters.

When a match is found, the sector number is added to the log. If the Stop on each match is enabled, a dialog box will be displayed with the sector number. The log can be viewed at any time, and sorted on the status column to see if any matches have been found. A search cluster can be viewed by clicking on the entry in the log.

A possibly unique feature of the CnW Recovery search is the ability to search both standard and compressed NTFS clusters. To enable this mode it is necessary to set the following flags on the Image Option display

Read in cluster mode
Test for NTFS compressed clusters

It is also necessary to set the start cluster sector number, and the cluster size (typically 8).

As each cluster is read it will be tested to see if compressed. If compressed, it will be expanded and searched

Uni-code and straight searching

The option for uni-code search can be used on it's own, or with the standard search. Both little end and big endian strings are searched for

Limitations with raw disk searching

At first glance, searching sectors or clusters seems a fool proof way to find a string, but there are limitations that must be understood before using the results as forensic evidence. The two areas are fragmentations and logical file structure

If a string being searched for is at the end of a cluster, it is possible that the file is fragmented, and so the end of the string may be on a different cluster, not adjacent to the first. In this case, the string will not be found. Fortunately this is a fairly rare event. If the search string is 24 characters, and the cluster is 4K, then the chance of missing a string is about 0.5% (1 in 200). For a shorter string, the chance is misising it decreases, but then the chance of finding a string that is not relevant increases.

The second case where a disk search may fail is due to the data in a file not being as expected. It will be clear that if a string is contained within a Zip file, it may not be found as the file will not be opened. The latest Microsoft Office files are infact all compressed, and so strings will not be detected by a raw image search. Slightly less obvious is that some programs will in effect save every version of a file, (making Undos possible) and so the original string will be saved, but any edited version will be done with pointers. A raw search may find the original, but not a small edit of it. The edit could be a correction in spelling or a few words that are part of the search string. These may not be detected.

Optimising multiple search strings

It is extremely useful to beable to search for multiple strings in a single pass. Searching does have a computing overhead so it is useful to know that the length of the shortest string will affect the overall search speed. This means if you want to search for 'zz' it will be slower than a string which is longer.

Summary

The raw search function will normally find a string if it exists, but one has to aware of limitations. To help reduce these limitations it will be best to run multiple searches.

-0-

Recovering files from image format

[Home](#)

Once an image file has been created with CnW Recovery software, it is normally possible to recovery files from it. The reasons for creating an image can include the following

- An exact copy of the disk, as original disk to be returned
- An exact copy of disk, so that original disk is not changed or corrupted in anyway.
- A copy of good sections of a damaged or failing disk

For good copies, then recovery will proceed in the normal way, once the image file has been selected as the input drive. This will include redefining partitions etc.

To recover files from a damaged disk can require extra stages, or operations.

Logical recovery

Logical recovery will try and recover the file files by reading in the conventional way, either with a full recovery, or in a mode such as From Directory Stubs. If the image is very corrupted, then this recovery mode may fail or hang before all the files are recovered. If using NTFS, then it is worthwhile using the mode of Recover from MFTs, and select the MFT range. It may then be possible to extract the file in several attempts, missing out sections where recovery fails.

Raw recovery

Raw recovery is performed by using the [Image mode](#), and selecting Split on files. This should be considered a last resort mode, as typically file names are not recovered, but only file types. However, if the main reason for recovery is to extract photographs, this can be a very successful mode. Many photographs do not actually have a meaningful file name, and so there is nothing to loose. At times, the recovered file name will include the date the photo was taken, but this information is not always contained within the JPEG file.

As in logical recovery, it can be useful to select the range of the disk to be scanned.

Raw recovery does have a major limitation in that it will join together fragmented files. For photos, they are normally fairly small, and so do not get fragmented. For a multi GB file, the chances are extremely high that there will be some fragmentation, and this will result in a corrupted file.

The types of file that are recovered are described in the [Raw files](#) page.

The number of files does increase on a regular basis.

Shadow disk

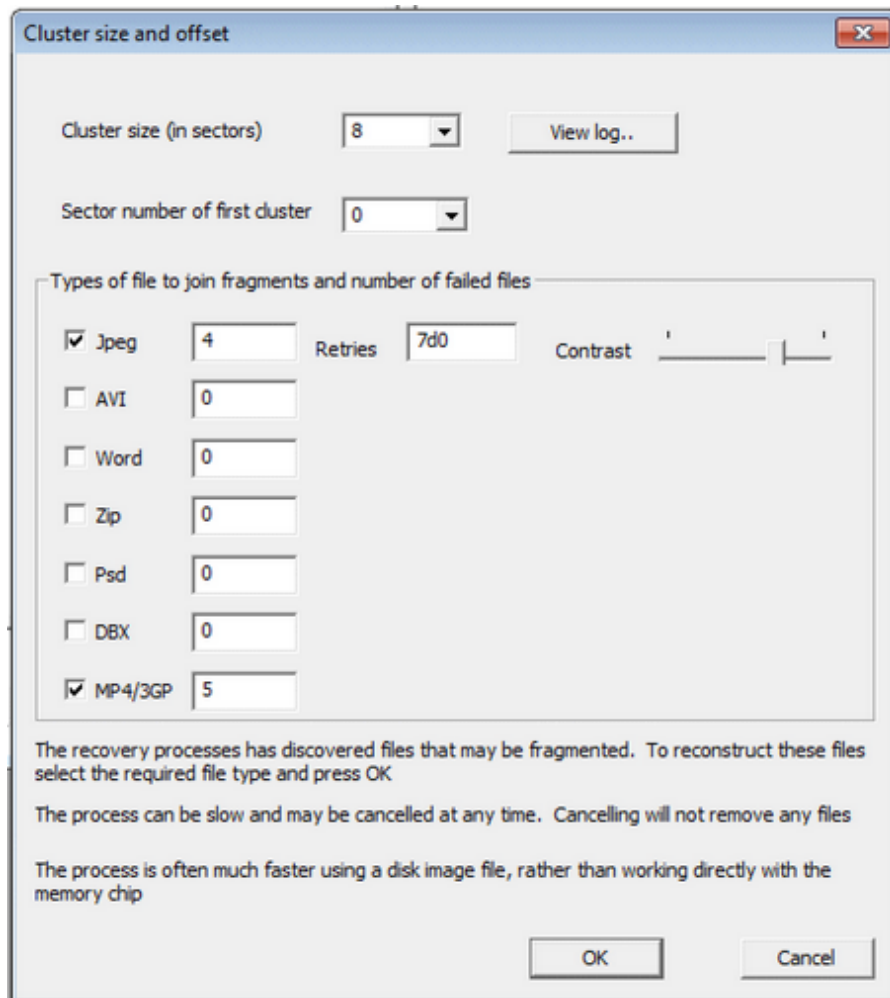
When an image files is selected, there is also an option to enable a shadow disk. The reason for this is when an image file has been created, but has missing sections. An image file should have as complete a directory area as possible, but then only the areas where files are stored need to be added. By using the shadow disk, areas of the disk that have not been imaged, will be read from the shadow disk. The shadow disk will only be accessed once for each required sector, so a failed sector will not be read many times. This will mean that although recovery will not be complete, it will not be exceptionally slow.

-0-

Fragmented file processing

[Home](#)

When the Process Fragments option is enabled, once the disk has been scanned, the option box below will be displayed. This will run automatic data carving routines.



The box indicates the number and type of files that could be processed. ie it will try and determine the location of each fragment of the file, and reconstruct the file. This process has a variable success rate, and can be slow, but will often reconstruct files that have otherwise been totally lost.

To assist in this operation there are two very critical values that need to be set which indicate the original size and location of clusters on the disk. See the section on [disk clusters](#) for more details. The example above is from a small memory stick, and so a cluster size of 4 has been detected. For most current hard drives, the most common cluster size is 32 or 64 ie 16K or 32K.

The option box above does have a link to the log which can be viewed to help

assess the correct cluster sizes. On a small disk the calculation of cluster size and offset is often correct. On a large disk, and in particular one that has been heavily used, it is common to require manual setting of the cluster sizes. It is often very useful to examine the file starts of the files to be processed, such as JPEGs. If all such files always start with the same sector offset, and multiple of increment sizes, then this is the best value to use.

JPEG Options

JPEG fragmentation can be complex, and so there are two additional options to assist with such files. The retry count and contrast value. There is no correct setting for these values, and sometimes trial and error will be required. The JPEG routine works by searching for a possible cluster that contains compressed data. This is then appended to the current partial photo and the result tested to see if this is still a valid partial photo. Each one of these is a retry value. The more retries, the chance of recovery improves, but the process becomes slower. As not all photos can be recovered, as the fragments may no longer exist it is important to have a cut off point of possible clusters. The default value is about 2,000 tries.

Once a cluster has been appended to a partial photo, tests are run to see if the photo is still valid by looking for a jump in the image. Visually this is easy to see when the bottom of the photo does not match the top. CnW works on the same principle but as a help, the contrast of the photo can be added. For images with a very low contrast (pastel colours etc) the routine will look for a very close match between sections. For photos with lots of contrast and images, a higher level may be required. If the contrast level is set too high, then a mis-match of photos may occur. If the level is set too low, no matches will be made. This an area of continuous development so later program updates may manage better results.

AVI

AVI is a common video standard, often used on cameras (rather than camcorders). Some cameras record the data in such a way that the video data is physically stored first, and then the header information is stored in sectors after the main data. Normal carving will fail, but the CnW fragmentation routine will detect this and correct the data.

MP4/3GP

MP4 covers a whole range of similar files for video, including many used on mobile phones, and .mov files. The defragmentation operates in several different ways depending on what data is available. The files start with a 'ftyp' segment followed by a 'mdat' and 'moov' segment. However, the order of the segments is not fixed, and so may be

ftyp-mdat-moov or ftyp-moov-mdat

The mdat segment contains the video data, and the moov segment all the control data and meta data.

A file will work with a partial mdat, but must have a complete moov. The moov segment contains many pointers that are used to try and select the correct fragment from possible mdat clusters. In a similar way, a moov fragment can be discovered and added to the file, with padding as required to ensure it is in the correct location. Development is underway to create missing moov segments when totally missing. For more details on 3GP processing, [click here](#).

AVCHD and MTS

AVCHD is a popular high definition video format used with many new video devices. The data files are .MTS. Such files can be viewed using Windows 7 media player. Typical data carving generates many MTS fragments, and this process will join many together. However, for a camera memory chip, the preferred method is to use the dedicated [wizard function](#)

Word

Word processing is for pre Word 2007. The success rate is limited when a device contains many word documents as it is very easy to obtain a false positive match.

Zip and DOCX

Current Office 2007 and later files are in effect ZIP files. These can now be processed and very high success rate has been achieved.

-0-

Jpeg images and metadata

[Home](#)

The JPEG standard is more than just compressed images. There are several sections in the standard that allow for application specific headers. These headers are used by many programs, such as photoshop, and by the camera directly to store information, or metadata about the files.

In raw recovery mode, CnW will examine this metadata and add fields to the reconstructed filename.

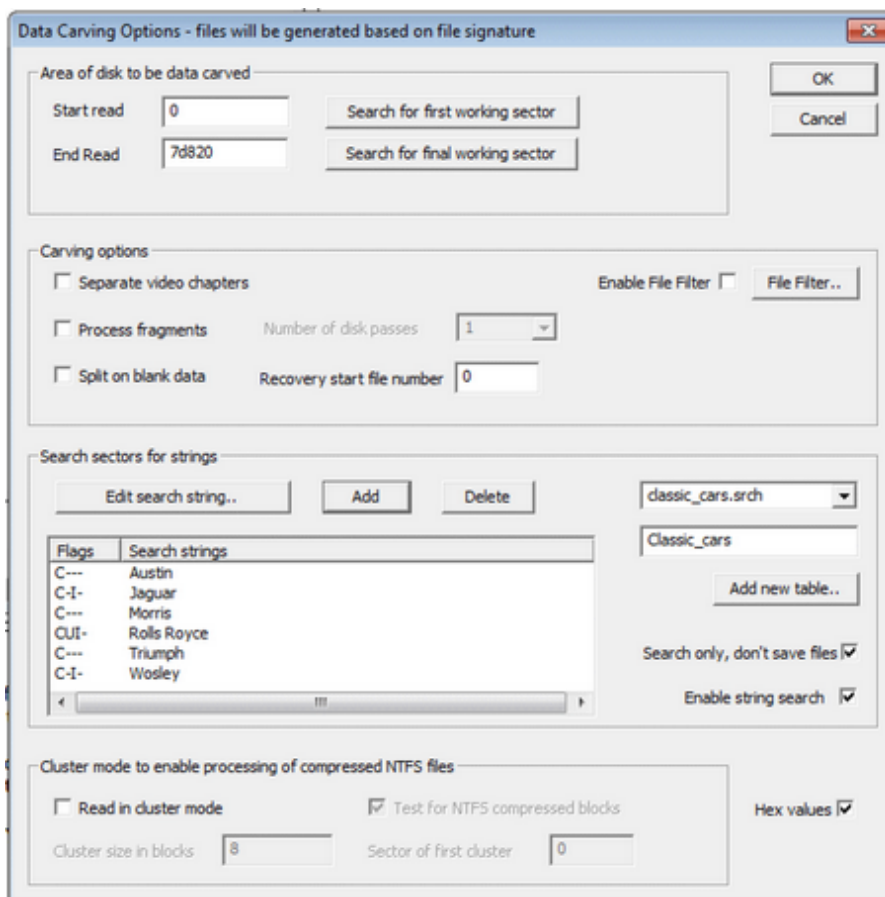
-0-

Fragmented Files

[Home](#)

CnW has several tools to assist with fragmented files when data carving. Some are part of the standard data carving, and others are special wizard function, mainly for video recovery.

A common problem with Raw recovery is that files are may have been fragmented. On a camera memory chip this is often due to photos or videos being deleted on a one by one basis, rather than a complete clearing or reformatting of the chip. When files are deleted separately, the space they used to occupy is used on new photos. Each photo is a different size, so sometimes a new photo will occupy multiple gaps, and is a fragmented file. For normal reading, the file allocation table (FAT) takes care of this fragmentation and so it is not a problem.



Raw recovery of a file is necessary when the FAT, or directory information is missing or corrupted. On camera memory chips, it can also be because all files have been shifted a few sectors, normally due to a software glitch somewhere.

To enable this option, you need to check the boxes Split on possible file starts and Process fragments. The routine is currently fairly slow, and only works when file fragments are actually sequential on the drive / memory chip. It works best on camera memory chips. Once the Process Fragments has been checked, a normal recovery of files is done, followed by automatic fragment processing. The process can be cancelled at any time.

There will be times when fragments of files will exist, but certain fragments have been overwritten. These images are impossible to recover. The aim of CnW Recovery, is to recover files that could have been read if there was a valid FAT. This will include files that have been deleted, but not overwritten.

Typical success rate of recovering fragmented jpegs will be approx 25-75% of images that first appear incomplete, though it is media dependant. It should be noted that very few recovery programs attempt to recover fragmented files when operating in raw or image mode.

For fragmented video files the best solution is to use one of the wizard functions, ie [3GP/MP4](#) or [AVCHD](#). These are optimised for complex recovery of deleted disks

-0-

Fragmented AVI files

[Home](#)

The two most common files on a camera memory chip are JPEGs and AVI files. Typically, memory chips do not get fragmented, but if the chip is full, or individual files have been deleted, then fragmentation can occur. As long as the FAT file system remains intact, there is not problem. If the chip is deleted, or formatted, then all details of file fragments is lost.

Using signature analysis, many files will be recovered, but the fragmented ones will be lost. An AVI file has a very strict structure, and it does rely on the final 16 bytes of the file being intact, and in the correct location. In other words, a 99% complete AVI file will not read - although a repair would be possible.

Once the CnW software has done a recovery of all files on a memory chip, the files are tested to see if valid. At this point, it will try and reconstruct non valid AVI files. The AVI recovery process has two possible approaches, probably unique to CnW Recovery software

Approach 1 AVI recovery

The first approach to recovery of fragmented AVI files is to search the disk for the index section of the file (idx1) stored at the end of the file. Once this has been located, a list of each file chunk is known, along with it's size and location. It is then possible to test possible clusters of data to ensure they fit with the index. By working this way, it is extremely unlikely to get a false positive match for a AVI cluster

Approach 2 for AVI recovery

Not every disk will contain a full AVI file, as some fragments may have been overwritten. In this case, if it is not possible to recover the complete index, a new one will be created based on the amount of the sequential avi data that can be read. The file will not be complete, but it can be viewed.

For more details, see the section on [Fragmented JPEGs](#). and ensure that the Process Fragments is enabled.

-0-

Data Recovery Tutorials

[Home](#)

This section is designed to assist users with common data recovery procedures. It gives a step by step guide and highlights certain approaches that can be used with different types of data recovery.

For many data recovery requirements, the wizard will guide the user through all stages until files are recovered. For more complex recovery requirements, it is necessary to use the 'Manual' mode. This page points to many common scenarios for data recovery.

It is always worth while starting with the [Wizard](#), as it will do a simple [media test](#), and a simple [logical structure](#) test. These tests will give a very good indication of whether the media is physically sound, and logically sound, or if there are detected problems.

Stage 1

With any recovery it is essential to know if the media being read is physically OK. If the Wizard test comes up with physical errors, or there is any concern about the drive, then it is best to make an image of the drive. The major upside of this is that bad sectors are only read once, and so the recovery procedure is much much faster, and if the drive is failing, then as much data as possible is preserved for logical recovery. The downside is that a data area, the size of the drive is required to make an image. Thus a 320GB drive will require another drive of at least the same physical size. The image is made as a file, and so can be on any logical storage device, such as a local drive, USB drive, or an external RAID. For details on imaging, see the chapter on [Image and Raw Recovery](#)

Stage 2

The next stage is to determine, if not known, exactly what type of media is being handled. For disks it is normally an NTFS, FAT, or HPFS (Mac) disk. This is typically controlled by the boot sector, and partition table

Partition and boot sector problems

- o [Partitions and analysis](#)

FAT, NTFS and Mac

- o [Recovery of lost files on an otherwise working disk](#)

NTFS recovery

- [General NTFS recovery](#)

HP Mediavault

- [HP Mediavault](#)

Camcorder

- [Recover from video camera with a hard drive](#)

Photo recovery

- Photo recovery from a memory chip

Disk imaging

- [Image failing drive](#)

-0-

General NTFS Recovery

[Home](#)

There are several common problems with NTFS disks. Most are related to failed sectors in the boot area, or at the start of the MFT. All can be recovered from, often with a high degree of success. If there are more than a few bad sectors, it is always worth [imaging](#) the disk first.

Stage 1 - NTFS Recovery options

To start recovery, in the main menu select Recover and the NTFS menu will be displayed, as below.

Recover Options NTFS

☒ Full recovery
 ☐ Recover deleted files
 ☐ Skip known good files

☐ From directory stubs
 ☐ Scan all MFT entries

☐ From File entries
 ☐ Select MFT range

☐ Recover unused space
 ☐ Recover slack space

Static

Start scan: 3f 0, d8f90 0, 587e290 0, 587e2cf 0

End scan: d8f51 0, 4a85300 0, 3c90231 0, 3c901f2 0

MFT Start cluster: 15685 0, 40000 0, 0 0, a755 0

MFT start sector: ab467 0, ff6f90 0, 587e290 0, 58d1d77 0

MFT entries: 5e0 0, 123a0 0, 0 0, 2d50 0

Cluster size: 8 8, 8 8, 8 8, 8 8

Display MFTs

Display in Hex: ☒

Serial Number: coddc327

Analyse disk...

Output destination: g:\jr3\job15\

Abort file copy on bad sector: ☐

Browse... Enable ☐ File Filter...

It is important to review the values that have been filled in. The display above is for 4 partitions, but many disks are just a single partition, in which case only the first of the 8 boxes will be non zero.

The most important values to enter are the Cluster size, and the MFT start cluster, which can be used to create the MFT start sector. One way to fill these in is to run the Analyse Disk function that will scan the disk to find the first long run of MFTs.

The number of MFT entries is not critical, so can be set to a reasonable size number, eg 64000 or 0x10000.

If the above values are wrong, no damage will be done to the disk drive, but data may not be extracted, or not recovered correctly.

Stage 2

The next step, once the parameters have been set is to determine the recovery mode. There is no correct answer, but the two main options are Full Recovery or Recover from file entries. The Full recovery is normally used when the file system is intact. In this mode, the recovery program emulates the operating system, and will follow the directory tree. A common error that is displayed when using this mode is a message that the INDX is not found. The solution for this is to use Recover from file entries.

If Recover from file entries is used, there are then two useful options.

Scan all MFT entries. This is a mode where the whole disk will be scanned for MFTs. This can be slow and if it is canceled part way through, the entries that have been found can be used.

Select MFT Range. This option has two uses. Firstly, if it is known that the file has an MFT within a certain range, it is not necessary to read the whole disk. Secondly, a limited range for recovery can be selected. This could be after the program has a problem, hangs etc trying to recover from a particular MFT. If for instance there is a problem when recovering a file with an MFT of about 1,000, then a new attempt could be made starting at say 1050. The option can also be used when it has been necessary to cancel part way through a recovery. Recovery can then be started where it was terminated.

Stage 3 - options

For each recovery mode, there are several options that may be applied. The most popular will be Recover deleted files. NTFS marks a MFT with a flag to indicate that the file has been deleted. It does not guarantee that the data is still available, as the data area may have been overwritten, but it does retain the location and details of the file. CnW software will place all such recovered files into a directory called Deleted, but the directory structure will remain intact.

-0-

Recover video from camcorder with a hard drive

[Home](#)

Many camcorders today use an internal hard drive, rather than removable DVDs. These hard drives are typically FAT32 and so recovery is fairly straight forward. Probably the most common reason for data loss is accidental use of 'Delete All'.

The main area that needs care with when dealing with a video camera directly is accessing the drive. For many cameras, when they are plugged into the USB port, they should appear as a logical drive. If it is not possible to access the drive as either a logical drive, eg Drive F: or a physical drive, eg Phys-2 then CnW Recovery software will not be able to assist.

The best advice at this stage is to create a image copy (Use Image raw function) so that any accidental use of the camera will not lose any more files.

There are two procedures that can be followed to extract your video files

Deleted file recovery

Most video recorders use FAT32 as the disk file structure. The first approach to take on recovery is to [read the disk](#), and enable recovery of deleted files. This is the preferable approach as filenames will remain intact. If sections have been deleted, and then new ones filmed, the resulting files may not be complete, or fragmented, in which case some errors may be expected. However, the vast majority of video will be recovered.

Raw Image recovery

The [Raw Image](#) approach should be tried if the deleted file does not recover all the relevant files. Once the files have been recovered, as MPEGS, it may be necessary to convert and merge them to make a viewable video disk. Details are in the chapter on [Camcorder recovery](#).

-0-

Recovery of lost files on an otherwise working disk

[Home](#)

A common problem with many disks can result from 'operator' error. A typical scenario is when a disk is repartitioned, or reformatted. The end result is a working disk, but with some or all files missing. CnW Recovery can help.

One approach is a logical read followed by an Image Raw to find files in unallocated space and a deduplicate. There are two straight forward stages to this process.

Stage 1

Read the disk with the standard recover function but select the 'recover unallocated area'. In this mode, the program will first read all the files and internally record the locations that they are stored in. The second stage is that it will perform an Image Raw, and extract files, but ignoring any area of the disk that has been previously read. The result is the !recover directory just contains files from the unallocated area of the disk, which will represent all possible missing files. NB, there will obviously be problems if the original files have been fragmented.

Stage 2

It is very common to find old copies of files within the unallocated area. This is where the deduplicate function is used. By selecting the log and the DeDup function, all duplicate files will be removed. The program works so that in preference it will remove any file that was read from the unallocated space, and retain files read in the main file system. The final result is that only one instance of any file will remain. It should be noted that some program files are actually stored in multiple locations, so do not run this function on a working disk image, but only when trying to process data files.

To assist in recovering may be just photos, the file filter is a useful option. It can be selected so that only JPGs are recovered. This procedure is totally compatible with the method outlined above.

-0-

Photo recovery

[Home](#)

A very important type of data recovery is to recover photos, typically from a camera memory chip. Failures can happen for several reasons, but the most common for camera memory chips is corruption when the chip is transferred to a PC for reading. It is very common for critical FAT information to be overwritten, or corrupted.

Photo file names

It is fortunate that photos do not require a file name to be useful, and the majority of users just rely on the sequential name the camera allocates to each picture. It is always best to try and read the memory chip logically, and so repairing the FAT and control information may be required, or alternatively, just an Image raw read will recover photos. In both cases, it is possible that a photo may have been created in multiple fragments, and this needs to be solved. The other reason for recovery is due to accidental deletion. Solutions to all these problems are described below.

Photo Wizard

The easiest recovery solution is to use the [Wizard function](#). The wizard will analyse the memory chip and determine the best way to recover the photos. When in the recovery procedure, thumbnails of photos will be displayed to give confidence that recovery is possible and progressing. For the demo, the thumbnails will be displayed, but no photos are actually saved.

Raw recovery and data carving

Use the data carving routine and this will find all JPEGs, and save them in a jpeg directory. To assist with disks that have thousands of jpegs, the directory is limited to 5,000 images until a new directory is created. To help with identifying images, when ever possible, CnW will add the photo date and camera to the file name, along with a unique incrementing number.

-0-

Imaging failing drive

[Home](#)

Many disk drives fail in a small area, or become very slow to read. For these drives it is best to create a disk drive image as soon as possible. The technique below is designed to put as little stress on the drive as possible. However, there is always a danger that the drive could fail totally at any time. If the data is very critical it may be the time to consider a hardware drive repairer, rather than risk this software solution.

Stage 1

The first stage is to try and establish how valid the drive is. A very simple approach is to select the drive and use the View function to look at areas of the disk. Can it read sectors near the start? Can it read sectors near the end? A sector that takes a long time to display indicates it is near failure point. A sector that displays 5A 5A 5A ZZZZZZ has failed, and cannot be read.

Stage 2

Determine the type of disk. Main disk types are FAT (normally external drive), NTFS (main Windows disks) and HFS+ (Macintosh). They all have slightly different optimum ways to be imaged.

Stage 3

Set up CnW to save the disk image. For this you will require a logical drive with enough space for a file of the length of the disk to be imaged. Thus to save the image of a 1TB drive, you will probably require a 1.5TB NTFS drive - or a network drive with adequate space.

Stage 4

The most useful sector to image is the boot sector, sector 0. If this sector can be read, start a full image.

Stage 5

Watch the imaging and see if it goes slowly, or lots of errors are detected. If so, it is the time to consider cancelling and working on incremental imaging. Slow is when it can take several minutes to increment the sector number on the screen, this normally updates every few seconds

NTFS disks

The most important first section to image is the \$MFT. For single

partition disk, this will start at 0x60003F for XP and 0x600800 for Vista and Windows7/8

Mac Disks

The typical starting point is 0x64028. This is also the area where there is often much failure on Mac disks

Fat Disks

There is no typical directory space on a FAT disk, though stage 6 may help with the root.

Stage 6

How to find where the full directory is stored. At this point in the process it is necessary to switch between reading the physical drive and the image file. By reading the image file no stress will be put on the drive.

Select the image file as the input and then select Recover. An options box will be displayed that will give the start of the directory / catalog. For NTFS and MAC there is also an option to display the directory locations. This will be the next area to attempt to image.

Stage 7

Finding the location of files. This will indicate what area of the disk should be imaged. For this process to give accurate results, the disk image should be the size of the actual disk (other wise attempts to read past the end of the disk will give meaningless start sector values in the log). To pad the file, the final area of the disk should be imaged, even just the last 10 sectors will work. The padding may take time on a large disk.

-0-

Video file recovery

[Home](#)

A major class of data recovery is of video files. There several reasons for this - some listed below

- Video is created on mobile devices
- Often FAT32 file system
- Often uses removable media
- Operator error can happen
- Devices can be dropped
- Media is moved between devices and computers
- Devices may have been finalised

Any of the above can result in video being lost or corrupted. CnW Recovery software has tools to recover video from all media types and file types.

GoPro and Drones. Thses cameras typically record high and low resolution videos in a multiplexed stream. The best tool to recover these videos is GoproRecovery, a separate CnW program. As of December 2019, this can now be launched from CnW and will share the same licence or dongle as CnW.

The links below high give methods of how to receover from different types of media

- [Recovery from Mini-DVD](#)
- Recovery from memory devices

Details of how MP4 files are stored on FAT32 devices

- [MP4 disk layouts](#)

-o-

Video recovery from mini-DVDs

[Home](#)

Mini-DVDs are still very popular for video cameras. They will record 30 mins to 1 hour depending on resolution on a single 80mm DVD. The DVD can either be a DVD-R or DVD-RW.

Problems often arise when the disk is removed before it is finalised. The other major problem is when the DVD-RW get formatted by mistake -often operator error.

The CnW tools for this are in three parts

- Data carving to find chapters
- Merge chapters into a single DVD compatible file structure
- Burn a new DVD

These stages can be done by hand - or rather better there is a [mini-dvd wizard](#) function to perform the first two stages, and then optionally also burn a new DVD.

Mini DVD systems normally use MPEG-2 to record their video. The benefit of MPEG is that even fragments of video can be viewed with requiring and special meta data files. Thus even a badly corrupted or damage disk can often be recovered to a level that video can be viewed. However to view files on a video player, the MPEGs have to be processed and indexed. They are then saved in a specific directory structure as below

```
VIDEO_TS
  VIDEO_TS.BUP          // backup of video_ts.ifo
  VIDEO_TS.IFO          // index info for the complete disk
  VTS_01_0.BUP          // backup of vts_01_0.ifo
  VTS_01_0.IFO          // index info for all video cells and
chapters
  VTS_01_1.VOB          // the video info - in effected merged
.mpeg files
  VTS_01_2.VOB          // continuation, a VOB is normally less
than 1GB in length
```

File names are always upper case.

The CnW tool will merge mpegs and create the files described above

-0-

Video recovery from memory devices

[Home](#)

Most current video cameras use memory devices for storage. These can have a capacity from a few GBs upto 32GBs, though this figure will probably double every year or two.

Memory chips are often reliable, but have two major problems

- FAT32 chips when deleted lose all fragmentation details
- Chips can be corrupted when moved between devices - eg camera to main computer

With memory chip video, there are several types of video file formats and several variations. In particular, there are multiple ways that data is stored which often results in fragmented files.

The main types of video format are as below

- MPEG - see notes on [mini-DVD](#)
- [AVI](#)
- [3GP, MP4](#), Quick time format
- [AVCHD](#) - high definition

The 3GP format is very popular with phones, AVI with compact digital cameras, and AVCHD with new high definition cameras and video recorders. They all need recovering in different ways, and CnW has many tools to assist, in particular with the fargmented files. Cnw can also recover from some MP4 files that have not been finialised.

-0-

MP4 disk layouts

[Home](#)

With a FAT32 disk device, the logical format of a file is lost if the file is deleted. To recover or reconstruct the file it is useful to know how it was originally organised on the disk. A video camera has limited memory and so often the logical structure of the video file, and the physical layout on the disk are different. To make matter worse, there are several ways that cameras solve the problem. CnW recovery software has tools to help when files have been deleted - simple data carving is often not enough.

An MP4 file basically has three sections

- Header (ftyp)
- Video data (mdat)
- Index information (moov)

The header is always first, and each of the other sections starts with a length, then data. There is also a padding atom, called 'free'. This allows the above sections to be placed on cluster boundaries.

The reason for what looks like the rather odd layouts is the way video is created. The major part of the file is the video stream, which can be maybe a few GBs long. A typical approach therefore is to record this directly to the media. To make the file playable, a header has to be added, and also all the index and meta data (moov) fragment. Logically these can be written to the disk when the data stream is complete, and by manipulating the file allocation table, the logical sequence can be changed to be different to the physical sequence. When recovering via data carving, this process has to be reversed, along with checks to ensure that the correct header and moov fragments are added to the selected mdat segment.

The table below describes several variations that have been seen from phones and video cameras. The cameras listed only represent possible examples and will never be an exhaustive list. Recovery from these formats should be possible by selected the 'process fragments option' in data carving. CnW has allocated short cut names for these formats that are displayed as part of the wizard function

Samsung HMX-H300

CnW Name :

Logical structure on disk

FTYP-FREE-MDAT only a single cluster, last 8 bytes are the MDAT length and header

MDAT data - just raw video data, padded at end with a FREE

MOOV starts on a cluster boundary, and is just meta data and index

Physical layout

MDAT data - padded with a FREE

FTYP-FREE-MDAT

MOOV

ie the data has been recorded first, then the FTYP header and MDAT length added. Final cluster(s) is the moov data

Kodak Zx1 Pocket Video Camera

CnW Name :

Physical layout on disk

FTYP - FREE

MDAT

MOOV - FREE

Logical structure for reading

FTYP - FREE

MOOV - FREE

MDAT

ie data is initially recorded first, followed with no known length, then by length and MDAT, then MOOV. Logically, the MOOV is stored between FTYP and MDAT. The recovery wizard reorders the clusters accordingly

GoPro, GoPro Hero 3+ Black, video camera

CnW Name : FTYP_MOOV_FREE_MDAT

Physical layout on disk

FTYP, MOOV in first cluster

MDAT

MOOV, FREE after the MDAT atoms

Logical structure

FTYP

MOOV

FREE

MDAT

GoPro Hero-3 Black edition makes recovery hard due to recording low and high resolution at the same time, along with a thumbnail jpeg and information text file - a recovery nightmare! Files can be fragmented in over 100 fragments.

GoPro Hero 4 Silver

Physical layout on memory chip

FTYP,MDAT, - in the first cluster

MOOV

Logical structure

FTYP

MDAT

MOOV

As in GoPro Hero-3, the data on the camera memory chip has two video streams multiplexed on cluster boundaries. The file on the memory chip may be in sequence, but is not sequential.

GoPro - basic

Physical layout on memory chip

RVFR - start of video data

FTYP, MOOV

FREE

MDAT stored at the end of the final cluster in the moov atom

Logical structure

FTYP, MOOV, FREE, MDAT, RVFR

Fuji Film FinePix XP50

Canon EOS 600D, 700D, 80D, 70D and Rebel range

Nikon D5100

CnW Name : MDAT_FTYP_MOOV_FREE

Physical layout on disk

MDAT

FTYP, MOOV in same cluster

rest of MOOV followed by FREE

Logical structure

FTYP

MOOV

FREE

MDAT

Coolpix P330

CnW Name FTYP_MDAT_MOOV

Physical and logical sequence the same, but can be fragmented

.MOV

CnW Name : M4_MDAT_MOOV

Logical structure

MDAT MOOV

This is rather unconventional as it has no FTYP atom

.MOV file

CnW Name M4_FTYP_MDAT_MOOV

Logical structure

FTYP

MDAT - at the start of the next cluster

MOOV - moov may contain padding with free areas

Each atom follows the previous atom with no padding of cluster alignment.

An unfinalised file may be missing the initial ftyp, and all of the moov atom.

To recover the data, the disk must have sample of a working video from the same camera.

-0-

mp4_scan

[Home](#)

The file mp4_scan.\$\$\$ is a diagnostic file for internal use with CnW Recovery software. It is stored in the cnwdata\temp directory, and so deleted each time the program starts. The file stores the result of the disk scan, looking for various MP4 type atoms. Once the disk has been scanned, the results are then analysed to determine the type of video that the memory chip contains. It is then possible to determine the original order of the data, and hence reconstruct the video files.

The structure of the mp4_scan.\$\$\$ is subject to change (and so not published) but does contain information such as sector number, atom type, atom length, and offset of atom within the cluster. By e-mailing CnW the file, it may be possible to determine new disk layouts. This is most relevant for memory chips, rather than hard drives that contain videos.

The following descriptions are for the types of video that CnW currently recognises. The names are unique to CnW as CnW is not aware of any industry standard.

-0-

MP4 brief file structure

[Home](#)

The MP4 file is a complex structure, largely defined by the Quick Time structure published by Apple. This page is brief description of some of the key points in a file that may assist with recovery, often by our [GPR recovery](#) program

The overall file structure three main elements, Ftyp, MDAT and MOOV. This section just concentrates on the MOOV segment.

The MOOV fragment is made up of atoms, and the important ones are described below, with details of what function they perform.

The atoms largely fall into two categories, fixed parameters for the file, and specific pointers for each frame of data and audio. To reconstruct a moov fragment, both sections are required, but the exact location of each frame is essential.

There is a Track atom for each data stream, typically the first stream is video and the second audio

trkh

Contains information such as file times, duration, speed etc.

mdia

Media Data atom

stco

StartChunk offset atom. This is a very important table. It points to start of each video (audio) chunk.

For videos that are greater than 4GB, the tag is replaced by co64 and the offset is a 64 bit value

Knowledge the codec being used is required to recognise a start. It could be a string, or maybe just a length.

stsc

Sample to chunk atom

stsz

Sample size atom

-0-

GoPro video recovery

[Home](#)

The GoPro camera is a very popular device for action video, including diving, parachuting, cycling and even on remote control helicopters.

The video is recorded on SD memory chips, and typically are 32GB FAT32, or 64GB exFAT. The FAT32 and exFAT means that when deleted the exact location of each file fragment is lost. For many systems, files are store sequentially so this is not a problem, but for GoPro, the camera records two streams of data at the same time, a high resolution video, and a low resolution video. Hence, the files are always fragmented.

There are also differences between the Hero-3 and Hero-4.

Looking a physical memory chip, a Hero-3 has the data in the sequence of

MDAT - FTYP (on a new cluster)-MOOV. On the hard drive, the logical sequence is FTYP-MOOV-MDAT

Many so called data recovery programs try and associate the FTYP-MOOV data with the following MDAT, and not the previous one

GoPro Hero-4, Silver stores data slightly differently

FTYP-MDAT-MOOV

This is the same as the logical structure, but the MDAT data is all fragmented.

A pattern that may be seen on both type Hero for the MDAT could be as follows

<H><H><H><H><L><H><H><H><L><H><H><H><H><L><L><H>
>...

where <H> is a cluster of high resolution video, and <L> is low resolution video. The clusters are just a fixed size, and have relation to the start or end of a video frame. It is from this 'mess' that CnW has to recover the video data in separate streams.

The basic GoPro Hero has a third layout - and does not normally store high and low resolution files interleaved. This format is referred in CnW as M4_RVFR_FTPY_MOOV_FREE_MDAT. The RVFR marks the start of the video

data which is physically stored first. The actual 'mdat' marker is stored after the standard ftyp and moov atoms.

The CnW approach is to use the [MP4/GoPro Wizard](#) function. This scans the complete device and looks for strings of data that maybe starts of video and audio frames. It also looks for possible MDAT, Ftyp and MOOV strings (known as atoms). The next stage is to reconstruct the files based on information stored with the MOOV atom. There are several possible interactions for the reconstruction, but when a video has been recovered correctly, a thumbnail image will be displayed. This works in demo mode to give confidence that all will work.

Other software packages

CnW does not like to be negative about competing software packages, but when it comes to video recovery there are many (big name) programs that indicate they have the video file, but when a user comes to view it, the screen just remains blank. CnW normally recovers these files, and if there is ever a problem, will look at the video memory chip until the data can be recovered. CnW has an SFTP server to allow for large file transfers.

-0-

HP MediaVault data recovery

[Home](#)

HP Media Vault is a popular RAID system, with one or two disk drives. The major problem is at times the controller fails, and data can not be accessed by a normal PC.

Versions of HP MediaVault

There are two generations Media Vault. CnW will process generation 1 which is Reiser based. This has model numbers such a MV2010, MV2020. Generation 2 is based on LMV disk structure, with model numbers such as MV2120, MV5140, MV5150. This is not currently supported by CnW software.

Structure of MV Generation 1

The structure of the MediaVault is a Linux processor with data stored under ReiserFS. The configuration can be a single partition, as a RAID-1 or a series of disks, JBODs, with multiple data areas. A simple tutorial on reading the disks is in [tutorials section](#) of this manual.

For initial evaluation, the wizard function for 'Corrupted and drive and deleted files' does basic analysis of HP Mediavault disks. It will indicate if the drive looks as if it is one of a pair. Failing that, if recovery is attempted on a single drive, it is likely that an error message that says the Root directory can not be found, may be displayed.

Reiser File System

Reiser FS is a rather different file system to most common file systems. What we are used to in NTFS, FAT and other Unix systems is that a file starts at the beginning of a cluster and fills up clusters. The final cluster though may have between 1 and the actual cluster size of data. Thus there is normally wasted space on the disk. With small files, and a large cluster size (eg 16K) every file will always occupy 16K. NTFS manages slightly better in that a short file, maybe 500 bytes can be stored within an MFT entry, but it still means a file always occupies 1K of data space (the size of the MFT record).

The Reiser approach, although Unix related with iNodes etc is to fill every block(cluster) and normally only 8 bytes may be wasted. The design is such that the file system is very fast. Reiser 3 was the last version of the file system, and Version 4 is currently on hold as Mr Reiser is spending some time in secure accommodation.

Two issues that the Reiser make recovery difficult are data carving and deleted files, and this is described below.

Reiser and Data carving

Data carving works by testing the start of a sector (or cluster) for a recognisable file signature. It then, typically assumes that data will follow sequentially. For Reiser, often the start of a file may be the middle of a cluster, and so to detect it means examining the contents of each cluster. Although possible, this has not been implemented within CnW (yet).

Reiser and deleted files

When a file is deleted in NTFS, and FAT, the directory entry is marked as deleted. If no data has been written to the disk, then the file can normally be found, and often recovered (subject to fragmentation). With Reiser, the iNode associated with main file/directory set to null values, and so the type of file or directory is lost. CnW software though has a recovery routine that will determine the original values of an iNode based on certain remaining parameters. This is not an exact science, and so not all files will be found. However, it scans the disk drive and finds many files, and where possible generates correct file names, and often the correct path. Because the basic directory structure is missing, one small problem is that files can be recovered multiple times. It can therefore look as if for instance 100GB of data can be recovered from a 40GB disk. This may be annoying, but data recovery is possible.

-0-

HP Mediavault Tutorial

[Home](#)

The HP Media Vault comes in a few flavours. This page is to assist any user with either the demo or licenced copy to recover data from the disks. (For the demo, all processes are the same as the licenced version, but no data will actually be stored).

Stage 1 Configure hardware

The first stage is to configure CnW to read the disks and also determine if it is a RAID-1 or JBOD or separate disks. The easiest way to physically read the disk is in a USB caddy, or for multiple disks, 2 caddies.

The question of RAID-1 or JBOD can normally be determined by the

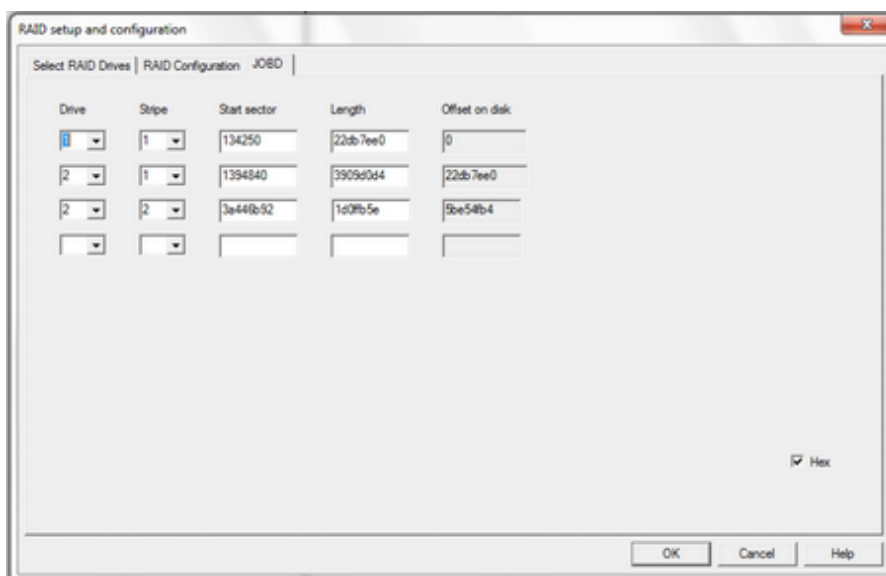
capacity of the Media Vault. If the capacity is the same as one of the disks, then RAID-1 is the most likely answer. If the capacity is twice the size of the disk, then JBOD was used. Apparently, RAID-0 was considered, but never implemented.

For RAID-1, only one disk needs to be considered, and so only a single USB interface required.

There is also a configuration of two single drives where the second drive is not logically related to the first drive. In this case the RAID option is not required.

Stage 2 Set up RAID if a JBOD

If it is thought that the disk is a JBOD, then the RAID option must be used. This is a chargeable option, so please contact CnW for details if not already purchased. For the demo system, just enter 'RAID' as the registration code. The registration screen is part of Configure in the Recovery functions and other options function. Please read the section in the RAID setup on [HP Media Vault](#) for details of setting up the JBOD. Hopefully, it will be done automatically with the analyse tool. A typical configuration is shown the screen below.



Stage 3 Confirming setup

This stage is optional, but often worth while to make sure that the disks are the correct format, and if relevant that the RAID is correctly configured. Using the Recovery functions and other options feature, the main menu will be displayed (ie this is not the starting wizard). The drive or the RAID must be selected at the top of the screen and the View function selected. This will display sector 0, the first of the disk.

For a RAID-1 disk, it should display the hex

```
00000000  42 72 6F 61 64 63 6F 6D - 20 4E 41 53 20 56 65 72   Broadcom NAS Ver
00000010  73 69 6F 6E 20 31 2E 31 - 20 4D 42 52 20 54 61 67   sion 1.1 MBR Tag
```

For a JBOD, viewed with 2: - RAID, the first sector will display

```
00000000  42 72 63 6D 53 65 4D 61 - 67 69 63 53 74 72 00 00   BrcmSeMagicStr
```

and in sector 0x80 of a JBOD, the data will start - this is the super block

```
00000000  50 A9 1E 0F B8 3E 89 00 - 83 5A A6 01 12 00 00 00   P@ ,>% fZ|
00000010  00 00 00 00 00 20 00 00 - 00 04 00 00 90 82 2C 39   ,',9
00000020  84 03 00 00 1E 00 00 00 - 00 00 00 00 00 10 CC 03   " i
00000030  4E 00 02 00 52 65 49 73 - 45 72 32 46 73 00 00 00   N ReIsEr2Fs
00000040  03 00 00 00 05 00 3E 1E - 02 00 00 00 31 4A 01 00   > 1J
```

An example of a stand along disk 1 - note the size at offset 0x28
0x3A251DE0 is roughly that of the size in sectors of the total disk, 500GB
Sector 1

```
00000000  00 00 00 00 00 00 00 0A - 00 00 00 00 00 02 00 00
00000010  00 00 00 00 00 02 00 0C - 00 00 00 00 00 02 00 00
00000020  00 00 00 00 00 13 42 50 - 00 00 00 00 3A 25 1D E0   BP :%à
00000030  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
```

An example of a stand alone disk 2- - note the size at offset 0x28
0x5743D224 is roughly that of the size in sectors of the total disk, 750GB
Sector 1

```
00000000  00 00 00 00 00 0F 42 4C - 00 00 00 00 57 43 D2 24   BL WCò$
00000010  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
00000020  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
```

Stage 4 Doing a recover

The final stage is actually recovering the data. For this stage, use the Recover function (in the main part of the program). On some configurations, the program may bring up the Partition screen with several partitions. In this case, select the line with Reiser FS. If Reiser FS is not shown, select Analyse partitions... and then select Search for existing partitions. Within a few minutes, a Reiser partition should be displayed, and at this point, the scan can be cancelled. Select the Reiser FS partition. Press OK

The next menu is the Unix Options menu. This allows for different types of recovery to be made, and the location where the files are to be saved (not relevant in demo mode). There are three main recovery modes

- Full recovery, this will try and read the file system in a conventional way, and recover files and directories. It is tolerant of file system errors, but will not find orphaned directories etc. This the fastest recovery function
- Scan directory stubs. This mode will scan all known iNodes for directory stubs. Hence it will find orphaned files from damaged

or corrupted disks.

- Raw iNodes. This node is the slowest, but can be extremely effective. The complete disk will be scanned, sector by sector to try and find any iNodes. This will then be noted, and the Scan directory stubs function called. This mode will find any lost files, and can also detect many deleted files. However, it should be noted that the directory structure may not be fully intact, though file names will be correct.

Once the mode has been selected, press Recover All or Select files.

Files will be saved in the output directory specified, and will retain the original directory structure.

Stage 5 Verifying files

The easiest way to verify files is with the log. A very good indication that a file is correct is to check the signature. For many common files these are known, and so should match, or at least match the family. For instance, .exe, .dll and other similar files all have the same signature. If no signatures match, there is likely to be a problem.

The demo does not actually save any files, so the log is the best guide as to what may be recovered. The display will show all files and directories as they are scanned. For more confidence, double click on a log line and the selected file will be displayed as a hex dump. (This does require the original disk to still be selected).

-0-

Forensic tools

[Home](#)

To do any forensic investigation, one must be able to access the media, and recover files from the same media. Investigation often goes further, trying to establish when files were written, which files have been deleted, or modified, and also what is on the disk, but cannot be seen by the standard operating system.

As a forensic investigation tool, CnW Recovery has a significant feature in that it will logically recover files from otherwise damaged or corrupt media. This will give the investigator many files that cannot normally be seen on the disk. In addition, files in unallocated space can also be recovered.

Although CnW Recovery software does not attempt to analyse file content, it will detect files that have been renamed to try and disguise the contents, in particular, most image files can be recognised by a signature rather than a (false) filename.

How each type of disk is analysed tends to be different, and so each type is described in sections below. However, common tools are based around the [log](#) which gives useful information on

- File name
- File size
- File dates, creation, modified, accessed
- Location of directory sector
- Location of data sector
- Number of fragments
- File extension
- File signature
- MD5 hash value (Forensic option only)

The [Forensic Report](#) (Forensic option only) does give details on operations and tests, along with many errors detected. This is generated in XML so that it may be included in a specific report on a particular disk.

A significant feature of using CnW Recovery software for recovery is that it does not use standard functions to recover files. The program is designed to be tolerant of disk errors, and hence also tolerant of deliberate changes to try and hide data. For instance, changing a boot sector will not necessarily allow a user to lock an area of the disk out. In this instance, it is also possible to modify certain parameters for a restore function so that for instance a large area of a disk could be examined.

The forensic option will include recovery of slack space for FAT and NTFS disks. For NTFS disks, this includes slack within the directory.

-0-

CnW Recovery forensic investigation tools

[Home](#)

CnW recovery software can assist in two main aspects of forensic investigation. These are recovering files, and tracking how and when they were written, changed or deleted.

Each type of media has it's own 'style' of information, so investigating a CD-R will be different to an NTFS hard drive, or a FAT memory stick. For rewritable media, there is often the issue of slack and unallocated space to be considered. A write once CD can be simpler, but multiple sessions add to the fun.

For all types of media, there are several areas that need consideration, but these can vary on type of investigation. Important points though are listed below

- All file names
- File attributes
- Dates that files were created, modified and accessed
- File signatures
- Hash values
- File integrity
- Which file a sector is part of

These features are all stored in the logs for each recovery job done. The above points are generally device independent and represent just the data.

File names.

The name given to a file is often a good guide to the file contents. File names are made of several parts, the directory path, file name, and file extension. Most people have some structure of where they store files, and often this is the default for the application that wrote the file. If users want to hide files, then placing them in different directories, or using different filenames can mean that a quick glance at the media will overlook such files. They can also be marked as Hidden files within the operating system

File attributes

Probably the most interesting attribute for investigation will be the Hidden attribute. A normal hard drive has very few hidden files, and they are normally protected operating system files. CnW Recovery will always copy all files, irrespective of their attributes. The file attributes

are stored in the log so hidden, and system files can be detected. Other attributes such as compressed, or archive are not normally very interesting forensically speaking.

Dates and times

Dates and times can be very interesting to examine. Exactly which dates and times are stored can be media dependant, but typically created, modified and accessed are interesting dates.

The creation date is when the file was first created

The modified date is when the file was last modified

The access date was when the file was last accessed

All these dates come from the PC clock, and are viewed in local time. There can be issues where the modified date is earlier than the creation date, which at first glance sounds rather odd. It can arise if a file is moved from one medium to another, eg copied from a floppy to a hard drive. Then the new file on the hard drive will have a creation date of when the file was copied, but a modified date of when the contents were last changed. If somebody is trying to cover up a change, it is possible to change the system clock, and modify a file, and possibly then change the system clock back again. To do this consistently is actually very difficult and this type of attempt may well be spotted by inconsistencies in dates, and maybe dates in logs, or when writing external media such as CDs.

File signatures

Many data files have a unique sequence of bytes at the start of the file. This can be used to see if a file is the correct type for the extension applied. For instance, all jpeg files start with the hex bytes 0xFF 0xD8 - after which there can be many variations. Thus if a file has a .jpg extension, and not the first two bytes, then either it has been renamed, or there is an error. Forensically, the opposite way around can be of great interest. A jpeg file could be renamed .dat in an attempt to hide it. CnW Recovery software always checks a signature on each file and it would therefore detect such a file as jpeg and this information would be stored in the log.

File validation

In certain modes a file validation routine can be run. Although it cannot handle all known variations of files it can indicate if the file is valid or corrupt. This should be treated as a guide, and not as evidence

Which file a sector is part of

If information is found in a sector it is useful to know which file it is part of. The search function in the log will allow the sector number to be entered, and it will display the file (or files) that the sector is found in. Multiple files will sometimes be found if one of them has been deleted, and the disk area reused.

-0-

Discover deleted files

[Home](#)

With many forensic investigations, a very important aspect is to discover files that have been deliberately deleted. Fortunately, deletions through the operating system typically just mark the file deleted, and make the space taken by the file available for new files.

The investigator then has several tools to discover the files, recover the files, and at times, can even work out when the files were deleted.

Stage 1

The first tool to use is a standard recovery routine, but selecting the 'Recover' deleted files that appears in the recovery menu. If there have not been many operations on the disk, since the files were deleted, this will recover the deleted files, almost certainly correctly. As more file movements on the disk have taken place, the chance of a file being overwritten increases. It should be noted that for FAT32 files, deletion often removes the original location on the disk, but CnW software has [functions](#) to assist with this.

Stage 2

When directories are deleted, the directory is marked as a deleted file, but there is always a danger that this entry will be reused, and so a logical parsing of the directory could miss a complete directory branch. To overcome this, it is best to try multiple approaches to reading the disk. For NTFS, use the option Recover from file entries which will scan for all possible files and directories. When a parent directory is not found because the directory has been deleted, a dummy directory name will be created. For FAT disks, the option Recover from directory stubs should be used. This will scan the disk for all subdirectory entries. One limitation is that if the subdirectory has been deleted, there is no way to tell how long the directory is, and so at times fragments of the directory may be omitted.

Stage 3

Some files, when deleted will in effect escape from the file system. For these it will be necessary to use the recover Unallocated Space option. This is used once the disk has been read, and then all the clusters that have not been access will be analysed for possible files. Being raw recovery, there are very few checks on the files, apart from fairly comprehensive file signature checking, and sometimes logical verification of the files.

Analysis

Once files have been recovered it is often worth investigating when they were deleted. For FAT disks, no such information is stored, but for NTFS disks there are dates stored which will indicate when a file was last changed which is stored in the Attribute time.

Another useful piece of investigation to work out on an NTFS disk what file overwrote a directory entry. If the NTFS recover routine cannot recreate a complete directory path, then it will create a `lost_dir_xxx` entry where `xxx` is the number of the expected MFT. By looking through the log for the MFT with the value `xxx`, one can see what has been written, and when, to delete the directory.

-0-

ISO9660 and Joliet investigation

[Home](#)

When investigating ISO9660 and Joliet disks there are several areas that may be of interest. CnW Recovery software will assist on giving details of the media, and also each session that has been written to the disk, along with dates and files.

-0-

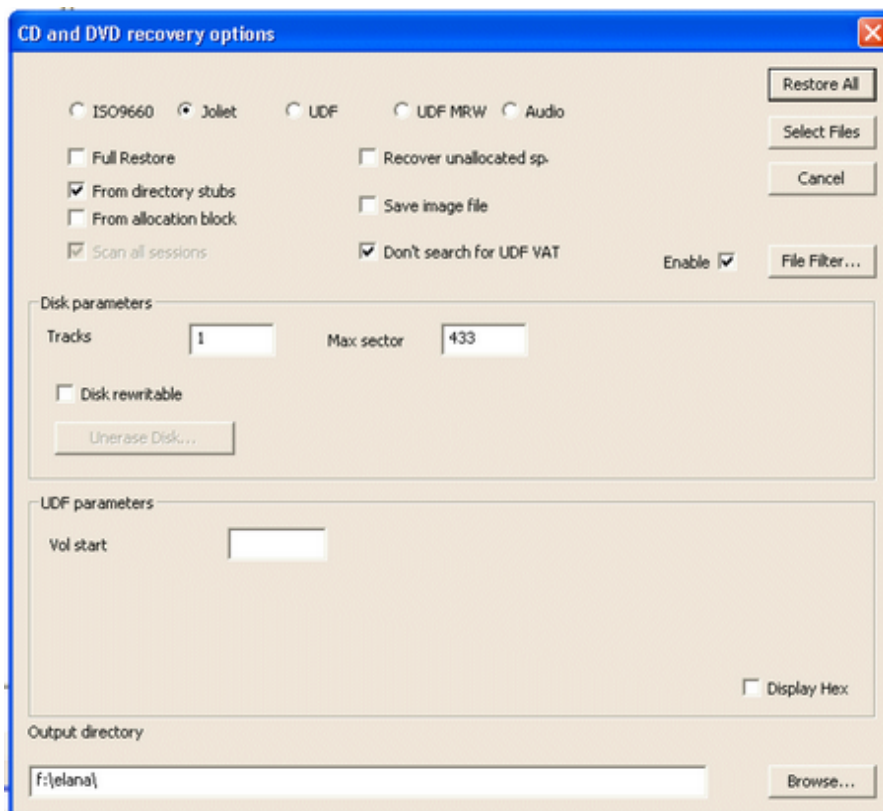
UDF forensic investigation

[Home](#)

UDF is typically used to record onto CDs and DVDs. However, it can be used on any media, and this includes Iomega Rev Disks. There are several principal versions, V1.02, V1.50, V2.50 and V2.60. Full specs are downloadable from the web. Large elements are also based on Ecma 167 standard.

UDF can be used on both write once, and read/write disks (eg CD-RW).

Forensically, write once disks are interesting because it is possible to in effect delete files, or edit files. Being write once, this is done by a slight of hand, which is virtual directories. Each time a writing session is finished, a table is stored at the end of the current data, which sets a logical map to the directory. This is the method that new directory entries may be made having new pointers to existing, or new files. Each new directory could be completely different, or only a minor change to previous directories, but can incorporate new files, or delete existing files. CnW Recovery software can reconstruct each session, showing which files were written, and when.



In order to view each session, the option box Scan all sessions should be selected - this is actually only enabled for UDF disks. The program will then search through the disk sequentially and find each UDF VAT (Virtual allocation table) and then will do a disk directory for each session. On a well used disk

there may therefore be the equivalent of maybe 80 tracks. Each track could be recovered on its own so file differences could be seen.

If all files are recovered then a significant amount of disk space may be required. At the end though, the DeDup function could be used to remove identical instances of any file.

-0-

NTFS forensic investigation

[Home](#)

NTFS is probably the most common disk format now used on a PC. In recent years it has become the default format, replacing the much simpler FAT32 format. It is a complex format supporting features such as compression and encryption. There is scope for users to hide data, and also scope for CnW Recovery software to recover data that is otherwise invisible.

The two most useful modes to investigate NTFS disks is to a full recover, and a scan of MFTs. The full recover, in particular when used with deleted files option, will show all the files on the hard drive, including recently deleted files. The scan MFTs will pick up all current files, and also files that have been left from a previous formatting of a disk. It can be very useful when an operating system has been reloaded, a many of the original files can still be recovered. An addition mode to the scan MFTs function is to scan the complete drive for MFTs. This will pick up more files, but sometimes the 'left over' MFTs will have rather odd subdirectory paths.

Features to assist with investigation

- Slack file recovery - for both files and directories
- Hashing of all files
- Full dates stored in logs for creation, modification, access
- The ability to discover which file a sector is used in
- Reads deleted files
- Checks file signature - useful when a file has been renamed to hide a file
- Recovers files even when the directory structure is incomplete
- Will scan disk for isolated / orphaned MFT entries
- Logs may be sorted in any date order
- Raw image scan of disk for unallocated space recovery
- MFT Parse** - view elements of the MFT
- Recovers registry files, eg NTUSER.DAT, logfile, \$usnjnrl

Third party tools

There are many third party tools to help with a disk investigation, some free, some chargeable. Some examples below. Can be downloaded from the web. CnW does not have any connection with these tools, and the information is just information

- RegRipper - Will expand the registry into a readable text report
- JohntheRipper Password recovery

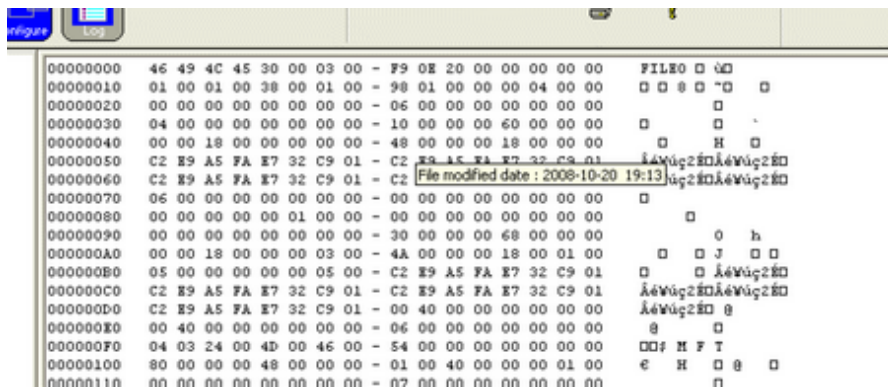
-0-

MFT Parse

[Home](#)

The \$MFT files is a list of all files on the pcurrent disk (or partition). The structure of each MFT record is well documented, but conatins many binary numbers and so can be difficult to interpret. CnW can be used to view an MFT sector, and when the mouse pointer is held over any part of the hex dump, appropriate fields will be explained. This will include file sizes, dates, as well attributes and pointers.

The same information will be stored in the log when a file is recovered, but the manual mode will assist with forensic investigation down the level of bits and bytes within the MFT record.



It can be seen in the screen dump above that the cursor is over the File modified date field, and so displays date and time.

The main sections of the MFT are all decoded, as follows

- 0x10 Standard Attribute Header
- 0x20 Non resident pointers
- 0x30 File name
- 0x50 Security descriptor
- 0x60 Volume name
- 0x80 Data run pointers and file size
- 0xA0 Index allocation
- 0xB0 Bitmap
- 0xD0 EA information
- 0xE0 EA
- 0xF0 Property Set
- 0x100 Logged utility stream

For the main header, typically the first 0x38 bytes, the following fields are displayed

- MFT header pointer to fix up : 0x30
- Fix up count : 03
- \$LogFile sequence number
- Number of times MFT has been reused
- Hard link count
- Real size of MFT record
- Allocated size of MFT record
- MFT value : 0x00 - this is the reference of the MFT in the \$MFT file
- Status flag indicating that the MFT is for a file or directory, and if used or deleted
- Fix up value, and it verifies that the value in offset 0x1fe and 0x1ff is correct, or incorrect

For standard Information, record type 0x10 the following fields are displayed

- Creation date
- File modified date
- MFT changed time
- File read time
- File sttribute, such as Read Only, Compressed, Hidden

For File name record type 0x30

- Creation date
- File modified date
- MFT changed time
- File read time

For data run record type 0x80 and 0xA0

- Offset to data runs
- Allocated size of file
- Real size of file
- Initialised size of data stream
- Cluster start of data runs
- Length of first data run in clusters - only the first is currently expanded

-0-

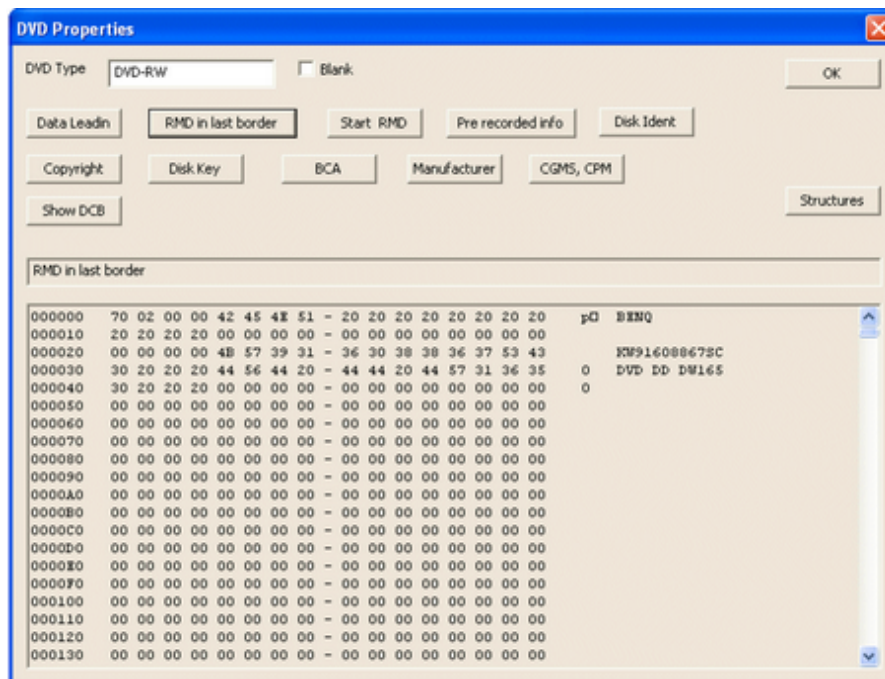
DVD Properties

[Home](#)

This tool allows the user to look at, and analyse parts of DVDs that are not normally seen. The functions are performed by the Read DVD Structure command on the disk drive.

The screen has several push buttons which are enabled if the actual disk drive enables the function. Not all disk drives enable all functions.

The display is a Hex dump, and it will be necessary to consult a drive reference manual to interpret the functions. For users with the forensic option, many of the fields are decoded, and stored in the Forensic Report log, and also included in the XML report. Basic descriptions are shown below.



Data Lead in

Gives details of the disk type, number of layers as well as start and end of data area

RMD in last border

Gives details of the disk drive that wrote the disk. This includes drive manufacturer, model and serial number. With this information it is possible to identify the actual drive used to write the disk. Forensically, it is therefore possible to tie up a DVD with a physical drive.

Start RMD

-0-

Data Carving

[Home](#)

Data carving is an important tool when attempting to recover files from either unallocated drive space, or from a disk that has become very corrupted. It is based on the CnW disk imaging routine, but goes several stages further and will attempt to automatically reconstruct files that are fragmented. It can be slow, but if a file is critical, it is well worth while, and quicker than trying to process by hand. Very few data recovery programs can recover fragmented files when the operating system details have been lost or corrupted - CnW often succeeds with this process. Forensically it logs the start of every file it finds, and the fragments when working on a reconstructed JPEG.

Recovery based on signature alone often works extremely well, but if a file is fragmented, then the recovered file will not be valid.

The approach that CnW Recovery takes is to do a standard raw recovery, based on signature and headers to track down sequential files. These files are then verified, and when the verification indicates a complete, and valid file, the space they occupy on the drive is marked as used. This helps reduce the number of sectors that have to be searched for other file fragments. Thus the program builds up an internal map of areas of the disk where fragments may be found, and areas where the data is known and in effect allocated.

The actual recovery of files is on a file type by file type basis. In raw recovery, the operating system gives no assistance as to where the fragments are stored, although the above procedures assist in helping lock out areas. Hence data carving does rely on a lot of trial and error.

Success rate does often depend on the mix of files on a disk or memory chip. If a single JPEG is fragmented, and all other files are XML files, joining fragments is easy. If there are a lot of Word Doc files, all fragmented, it is very easy to get false matches.

To reconstruct a file, first the starting stub is required. This is typically found by a signature, for instance a JPEG file always starts 0xFF 0xD8, 0xFF and then a 0xE0 or 0xE1. After the initial header blocks, a JPEG file is made up of sectors of compressed data. The recovery routine can therefore ensure that the possible sectors to add are compressed data, and then verify if the additional data still makes the incomplete JPEG file consistent. This process is continued until a complete file is constructed, or it is determined that this process is not working.

CnW Recovery software has automatic routines using data carving methods for files such as JPEG and AVI files. This list will grow on a regular basis.

JPEG Carving

JPEG carving is particularly useful with camera memory chips. Once a raw recovery is performed with a memory chip, often there are several images that do not open. If they are fragmented, then the process fragments function will normally reconstruct between 25% and 75% of these images. The routine works best when different fragments of the image are sequential, though not consecutive. If sections of the file are missing - because they have been overwritten - no data carving will work. CnW does not attempt to insert data to fix the image.

AVI Carving

AVI's have a structure that is very tolerant to carving. They are also tolerant to sections of corrupted data as long as critical pointers are valid. CnW software will therefore join fragments together to make a valid routine. Development is taking place so that when the end of an AVI cannot be found, a suitable trailer record will be added so that the reconstructed AVI will open and play.

DOC carving

Word documents do not respond to simple data carving techniques very well. There are several pointers in the file that can be used but the recovery rate using automatic routines is not very high.

PSD (Photoshop) carving

PSD files do have a lot of records, all with embedded lengths, and a simple 4 byte tag. It is therefore possible to step through the file validating, and predicting where the next tag must be. Knowing that tag must be at a certain location within a cluster, it is normally possible to locate the correct cluster

-0-

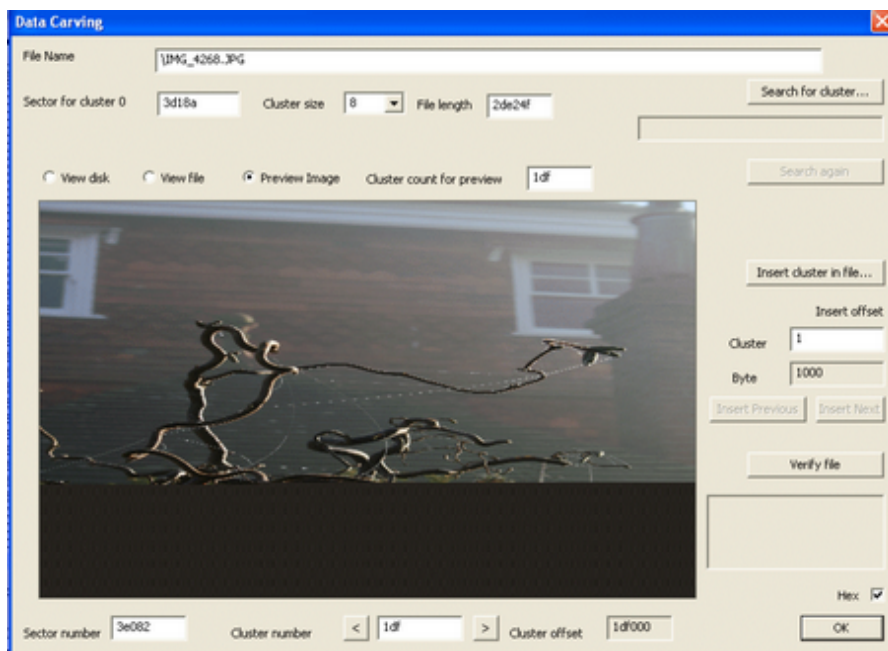
Manual Data Carving

[Home](#)

For investigation, automatic data carving is obviously the quickest method, but at times it is necessary to perform manual operations to obtain the best results. Typical occasions where automatic carving has significant problems is when a certain type of file has been fragmented, and there are many more similar files of the same type. A good example might be an Excel or Word file which all have key pointers to recognise file sections, but it is very easy to accidentally match fragments from multiple files by mistake.

The manual carving system allows for viewing and working with clusters, a file can be built up, or edited.

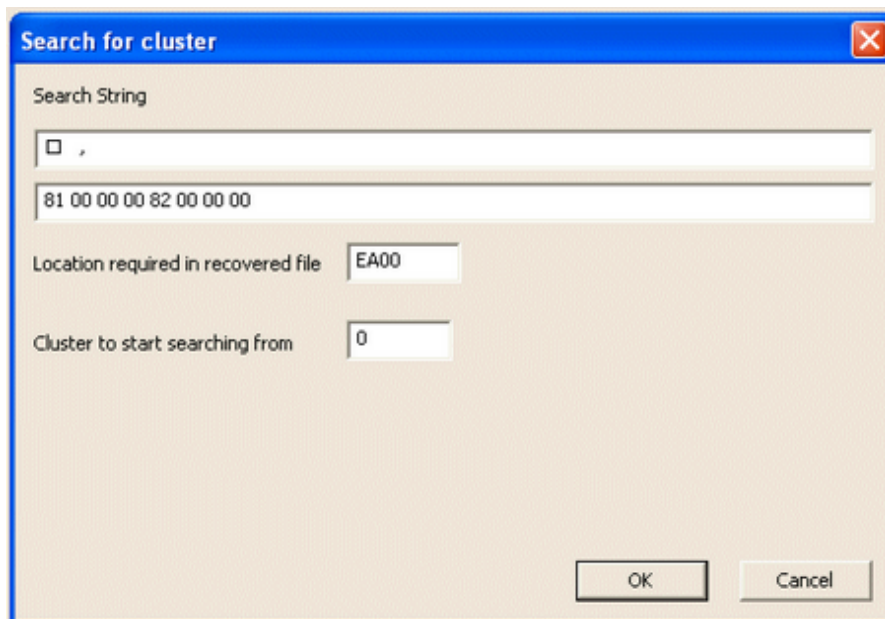
The options is part of the Forensic option only. To edit a file, the log must be displayed and the Verify Column clicked on. The Data carving option box will be displayed.



-0-


```
00000200 81 00 00 00 82 00 00 00 - 83 00 00 00 84 00 00 00    • , f "
00000210 85 00 00 00 86 00 00 00 - 87 00 00 00 88 00 00 00    ... † ‡
^
00000220 FE FF FF FF FE FF FF FF - FE FF FF FF FF FF FF FF
pÿÿÿpÿÿÿpÿÿÿÿÿÿ
```

In the example above, the cluster size is 4 (0x800 bytes) so the Root entry and first FAT pointer will be contained within the header. However, the second FAT pointer should be in cluster 0x1D, but as this was not the case it has to be searched for, by entering a location of EA00 and the values 0x81 0x82



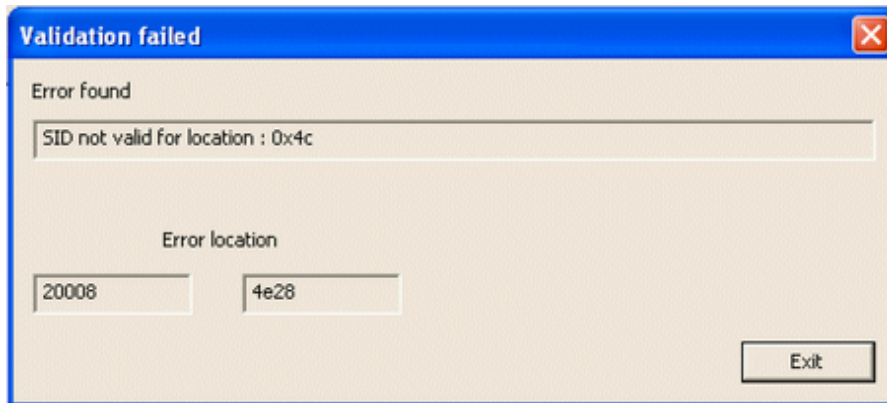
p219

-0-

File Validation

[Home](#)

The file validation routine will attempt to validate files for certain known types. The results will either be File Passed, or the dialog box below will be displayed.



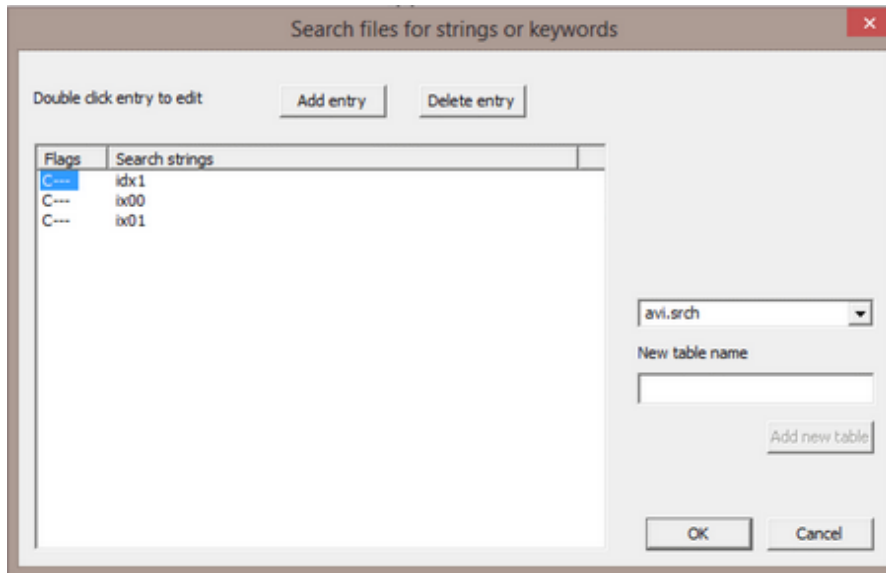
The error is format dependant and will try and indicate where the file failed in validation. The location, if known will also be set (in both decimal and hex). At the same time, the display will be set in the file view mode, and the fail cluster will be displayed.

-0-

Search for strings

[Home](#)

Search for strings function allows the disk to be searched for specified string either while performing a disk image, or as a stand alone function



The first stage is to enter a table name and the 'Add new table'. If a table has not been created, then the add and delete functions will not be enabled

Tables are stored and may be selected from the drop down combo box menu

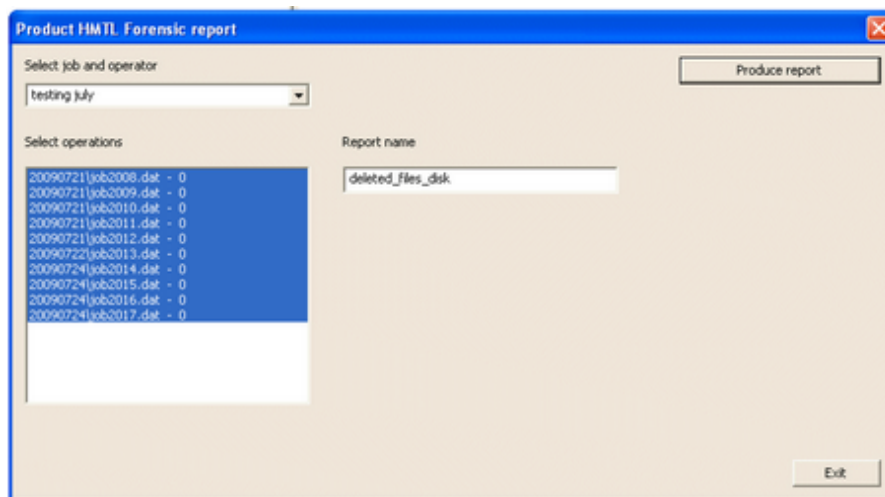
Strings may be entered with the Add entry function. Data can be entered in ASCII or HEX and there are options to search for normal strings, or uni-code. There is also a case ignore option

-0-

XML XML Forensic Report

[Home](#)

The forensic report produces a comprehensive summary of any recovery. It gives details of all operations performed on a disk. In a typical case a forensic recovery may consist of disk imaging, and also recovery in possibly more than one method. All these results can be grouped into a single, comprehensive report. The report is saved as an XML document (and XSL style sheet) so that it can be edited, and customised in Word, or just viewed using a Web viewer, such as Internet Explorer. To copy the report to a different PC, it is essential to also copy the `for_report.xml` in the style subdirectory.



The job name and operator are entered on the first CnW screen. It is a good discipline to set this up for each new job. However, as jobs can be selected by hand in the operations list box, this is not essential.

The details included in the report are as below

Disk report

The disk report is a very simple summary of the job. It includes the data and time of the operation, disk format, and recovery mode. The media type (eg disk file image) and the media serial number.

Disk imaging

For disk imaging, the start and end sectors are displayed, along with the MD5 hash value. This can be very useful if the image was not complete, and part of an incremental backup.

Disk recovery details

This section gives the basic details on the recovery. It includes which options were used, and basic file system parameters, such as cluster size and MFT start location.

Extension to signature match

This section displays all file extensions found on the disk. It then correlates them with the file signature found. For a good condition disk, for a known file such as .jpg one would expect all files to have a known signature. If this value is not the same it indicates either a different file type, or that the files have been corrupted (or maybe deleted).

This report will highlight files that have incorrect signatures for the extension. It could be a deliberately renamed file in an attempt to hide it. Thus if JPEGs were renamed .DOC, there would be a lot of .DOCs with failed signatures

There are three possible results of this test

- Signature match - the extension matches the signature of the file
- Signature different - the extension does not match the signature found. ie a file signature has been found, but not for this extension
- Signature unknown - the extension does not have a known signature

Deleted file overwriting

When a file is deleted the area can be overwritten by a later file. This part of the report will test each file where the signature and extension do not match and find if another file has been written over the start of the deleted file. If so, the deleted file name, overwriting file name, and the date of the overwriting file will be displayed.

To help keep the report a sensible length, .tmp files are not tested

Signature to extension test

This report is similar to the report above, but starts from the file signature rather than the extension. Again, it will be useful in finding renamed file extensions. If JPEG files were renamed .DOC, then this report would show a lot of JPEGs with incorrect signatures.

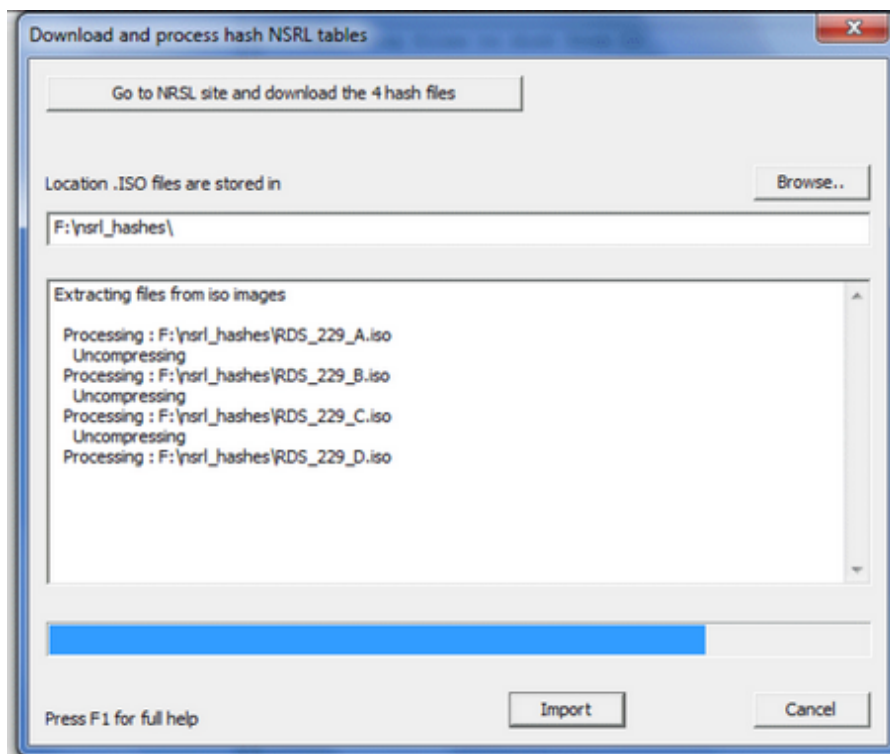
Keyword details

If keywords have been searched for the Keyword details give a brief summary. It will show the key word, along with the number of files it has been found in, and the total number of instances.

NSRL Hash tables

[Home](#)

The NRSL publishing extensive hash tables of known files. To assist with forensic recovery it can be useful to eliminate any known file that has not been changed. There is no point checking a Microsoft system file if it is exactly as it came out of the box. By checking the hash value it can be confirmed that the file has not been altered in any way.



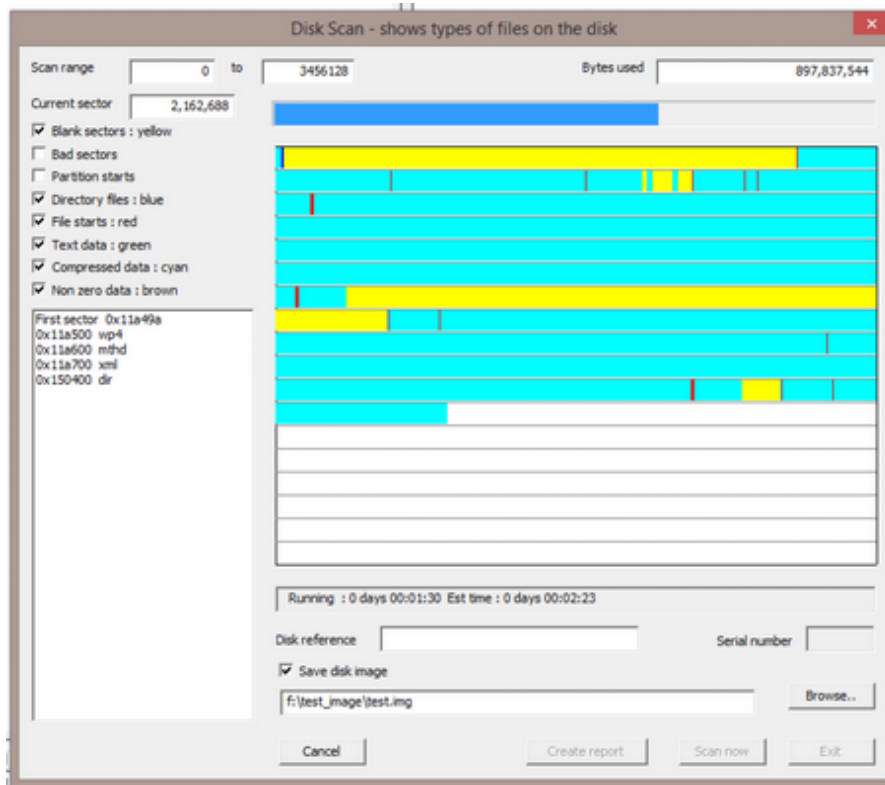
To generate a CnW recovery table it is first necessary to download the 4 zipped ISO image files from the NRSL web site. These are shown as disc_1, disc_2, disc_3 and disc_4. Currently they are named RDS_229_A (B/C and D). CnW will then read from the ISO image and unzip, and extract just the required MD5 hashed. This is followed by a sort procedure and then cleans up the temp files. The final stage will copy the sorted MD5 hash values table to the correct directory for selection within the [file filter](#). The data stored is sorted binary records, each record 0x10 (16) bytes long.

NB, the process does require about 5GB of free disk space. This tool is part of the Forensic option package.

Disk scan

[Home](#)

The disk scan function (forensic option only) will scan a disk, sector by sector and indicate where data has been stored. There are several categories of data it will detect, and can be selected by the user.



The serial number is the drive serial number as set by the manufacturer. This should be a unique number. The drive reference is made up of the drive name and serial number. This is used by default for the report name.

The types of sector detected are

- Blank sector - there is no data in this sector
- Bad sector, was unreadable
- Partition starts - this was a sector such as a partition sector, or BPB
- Directory files, eg a MFT entry, or FAT directory
- File start - has found a recognisable signature
- Text data. The data is largely text
- Compressed data. The data is largely compressed. This could be for JPEGs, video files, ZIP files, and sometimes programs, .exe, .dll etc

When the check box is selected, or changed, the display will be updated. This

way the colour of each data type can be seen clearly

When the cursor is clicked on the chart, the left hand box is filled in with possible file start data. In the example above it shows XML and WP4 amongst others. The number is the hex value for the sector analysed.

As an option feature when scanning, the a disk image can also be created. Unlike the standard disk image function it does not pad files when the sector range does not start at zero. It means that this method of imaging is not suitable for incremental imaging. As the function makes a lot of use of parallel processing, it is fast.

Create report

When the disk is scanned, the basic results are stored in the log - mainly in the forensic section. Also an XML report is automatically generated. It is stored in the log directory, typically `c:\cnwdata\reports`. The report can be viewed using Internet Explorer and the associated style sheet (`disk_report.xls`) is also required.

The report displays the basic information and (soon) will display images of the distribution of each type of sector found.

-0-

Virtual disk image

[Home](#)

CnW Recovery will read Virtual disk images, as part of the forensic option.

The Virtual disk image is selected as an image file and is recognised as the start sector has a signature 'KDMV', ie VM DK as little endian.

The routine in V3.62 handles disk images upto 2GB in length. This limit will be increased in later releases. The format works by having a look up table that allocates space to the data. Thus the VMDK image file may be considerably smaller than the disk image it represents.

The file format is defined by the standard VMware Virtual disks (format 1.1)

-0-

Forensic analysis tips

[Home](#)

Once a disk has been read, it is often required to do further analysis. The following notes are just a few possible tools that could be used to assist analyse certain files and logs. The ones mentioned are free and are included just because they have been used and seen to be useful. There is no relationship between CnW and the companies mentioned

LogFileParser - download from
<http://code.google.com/p/mft2csv/wiki/LogFileParser>

This program will produce a range of .CSV files for the log and User Journal
It works with files that CnW will recover and called and located as below

- LogFile typically c:\$LogFile

RegRipper

The program will expand into text files the structure of several system files, such as the registry (NTUSER.DAT)

-0-

File Selection

[Home](#)

The file filter is a group of functions to enable only certain files to be recovered, or to skip files not required.

[Overview](#)

[File extension selection](#)

[Date selection](#)

[Directory selection](#)

[File name selection](#)

[File selection based on MD5](#)

[File size selection](#)

[Import list of file names](#)

Coming soon - file content selection

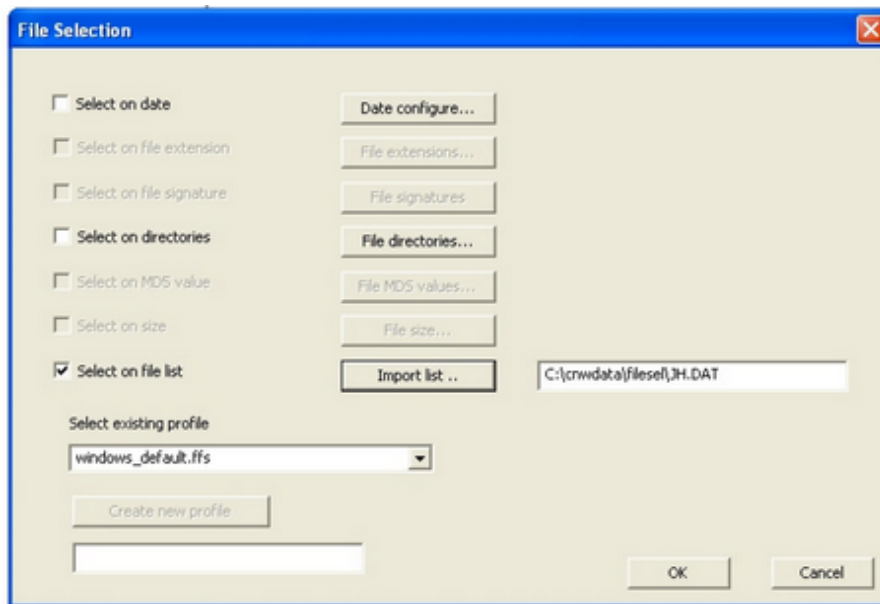
The File filter is an option of most recovery functions, eg FAT, NTFS, Data carving etc

-0-

Overview

[Home](#)

The file selection menu allows users to select files as they are being restored based on several criteria, such as date, type of file, file name etc. A file section may be made up of several different tests, so it could be based on a mixture of data, and file extension, and at the same time skipping very short files.



Users can create many file selection profiles, and use the relevant one for each type of job.

To create a new profile, just type the name required in the box underneath 'Create New Profile'. Once there is a name, then 'Create New Profile' may be used to initialise the file filter, and enable new entries to be made.

The operation is in two sections. First a detailed selection has to be made, by selecting the relevant configuration function, such as 'Date Configure...'. The actual function then has to be enabled by checking the relevant checkbox. In this way, filter routines may be built up and used in different ways. Thus it could be possible to have one requirement to select on directories and dates, but another job just wants the same directories, but no dates.

-0-

File extension selection

[Home](#)

A very useful feature when recovering files is to recover just the required ones, based on file extension. With this, categories of files can be either recovered, or skipped. Two simple examples could be to recover all photos, or recover all files that where not programs.

To recover all photos, either the JPEG option could be selected, or All Images could be selected. In many cases, the result will be the same, but All Images will also recover BMP files, and other known image formats, such as specialised camera formats

The Select all buttons are for quick selection of various types of files. The actual extensions they select are listed below

Select all images

- JPG and JPEG
- BMP
- GIF
- TIF and TIFF

Select all movies

- AVI
- MPG and MPEG
- MOV

Select all programs

- EXE
- COM
- DLL
- OCX

Select all sound

- MP3
- MP4
- WAV
- WMV

The lower list box is used to enter any other extensions, up to 15 characters in length. To add a new extension, use the Add extension button, and then enter the text.

Date selection

[Home](#)

These options will allow files to be selected by dates on any of creation, modified or accessed dates. There are 4 modes that data selection may be used, for each, or all of the three date types Date selection is based on days, any hours or minutes are ignored..

The screenshot shows a dialog box titled "Select file date range". It contains three sections for date selection: "Modified dates", "Creation dates", and "Access dates". Each section has three radio button options: "Date earlier than", "Date later than", and "Date between". In the "Modified dates" section, "Date earlier than" is selected with a date of 04/01/2006. In the "Creation dates" section, "Date between" is selected with dates 11/01/2006 and 29/01/2006. In the "Access dates" section, "Date later than" is selected with a date of 29/01/2006. The "Date between" option in the "Access dates" section is also visible but not selected. At the bottom of the dialog are "OK" and "Cancel" buttons.

- Date earlier than. When this flag is selected a file is selected if the date is earlier than the left hand date in the dialog box. Ie if the date displayed is March 23rd, then any file dated March 22nd or earlier will be copied
- Date later than. When this flag is selected, a file is selected if the date is later than the Late date, date. This is a late date is selected as March 31st 2003, all files of April 1st 2003 or later will be selected and copied

-0-

Directory selection options for data recovery

[Home](#)

This option is to optionally select the directories to restore, or skip. Any number of directories may be entered, and wild codes used to help define the directories. It can be very flexible, and it is not necessary to define the complete directory path, or file names. The use of '\ ' and '*' are as follows, best described by examples. All strings are case insensitive, but directories are shown in capitals just for ease of display

To copy or skip all files from a root directory ROOTDIR
 \ROOTDIR

To copy or skip all files beneath a directory WORKINGDIR
 WORKINGDIR nb, it does not start with a '\ '

To copy or skip files beneath directories containing the string CAT
 CAT

To copy or skip all files beneath directories starting with DOG
 DOG*

A directory name can be series of subdirectories, such as
 \ROOT\SECOND_DIR\WILD*

-0-

File name selection

[Home](#)

Files may be selected based on file name. With this option, the directory is irrelevant, although it will work in conjunction the [directory selection](#) routines.

File names can use standard wild characters, * and ? to match the required name

-0-

File selection based on MD5 value

[Home](#)

Files can be selected or skipped based on the value of their MD5 Hash. To use this function, a file has to be selected with a list of MD5 entries. Useful files could be ones that list all standard hash values for files within an operating system.

A very useful web site for these files is

<http://www.nsrl.nist.gov/Downloads.htm#isos>

On the structure of the files is an ASCII list of hash values, terminated by a CRLF - the file name is not actually relevant. CnW software will test, and if need be sort the file before using it, it is therefore possible to append multiple files together.

To automate this procedure, the [Hash Tables may be downloaded](#) and saved for CnW use

There are two options that may be taken when a file is detected that matches a hash value within the file, it may either be copied, or skipped. By using a hash list of standard operating system files, only changed, or user files will be restored. For a forensic investigation, this can save a considerable amount of time.

It should be noted that CnW Recovery software works with MD5 hashes, rather than SHA-1. Although it could be argued that SHA-1 is more secure, for 99.999999999% of the time, it is not significant. No known accidental clash to my knowledge has ever been detected.

MD5 Table structure

The MD5 file can be in two possible formats. The program will analyse the data and hence select the correct format to use. In each case, the table must be sorted with the lowest values stored at the start of the file.

- ASCII format. In this mode the file is entirely ASCII, with each line being 32 characters, terminated by CRLF. This is an easy format to generate from some published MD5 tables
- Binary format. In this mode the data is straight binary, with each record being 16 (0x10) bytes long, and no record terminator. The advantage of this record type is slightly faster running time, and a smaller file required of the hard drive. The NSRL download function produces this style of table, which is

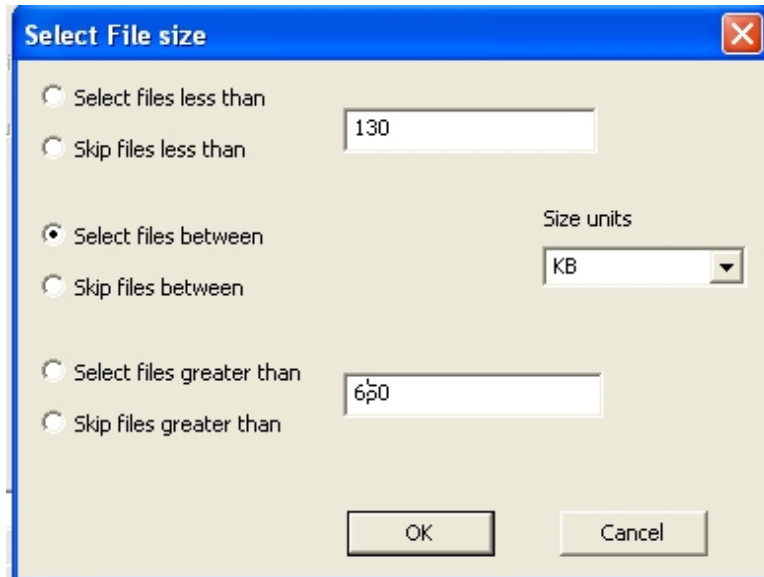
less than 300MB in length (Jan 2011).

-0-

File size selection for selective file recovery

[Home](#)

These options will allow files be selected based on their file size



All file size matching are actually done on selecte size in bytes. To make te entry of numbers easy, the units may be selected as Bytes, KB, MB or GB. If the size of units is changed, the display is updated to match the new units. Thus 2KB, when changed to bytes will display as 2048. Going the other way, 2000 bytes will display as 1KB, and is actually truncated to 1024 bytes

There are 6 radio buttons that will choose the mode of selection

Select files less than

Any file less than the size in the top box will be selected. If the box is displaying 1500 bytes, then files of 1499 bytes will be selected, but 1500 bytes will not be selected

Skip files less than

Any file less than the size in the top box will be skipped. If the box is displaying 2500 bytes, then files of 2499 bytes will be skipped, but 2500 bytes will not be skipped

Select files between

Any file greater than or equal to the value in the top box and less than or equal to the value in the bottom box will be selected. In the display above, a file of 150K would be selected, 800K would not be selected.

Skip files between

Any file greater than or equal to the value in the top box and less than or equal to the value in the bottom box will be skipped. In the display above, a file of 600K would be skipped, 100K would be selected.

Select files greater than

Any file greater than the size in the lower box will be selected. If the box is displaying 10000 bytes, then files of 10001 bytes will be selected, but 10000 bytes will not be selected

Skip files greater than

Any file greater than the size in the lower box will be skipped. If the box is displaying 12500 bytes, then files of 12501 bytes will be skipped, but 12500 bytes will not be skipped

-0-

Import List

[Home](#)

Import list enables the user to import a list of files to be selected.

-0-

RAID Drives

[Home](#)

Over the years several types of RAID have been developed. RAID stands for Redundant Array of Inexpensive Disks. The basic reason is to allow for a large amount of storage on disk drives, but to be tolerant of disk failure without losing any data. There is always a trade off between performance, cost and degree of possible failure. There are several basic standards, but also many proprietary variations. CnW Recovery concentrates on the most common standards with it's optional RAID recovery option. Please contact CnW for details of purchase. To evaluate the RAID option as part of the demo, enter the code 'RAID' in the registration code (rather than 'DEMO').

To use the RAID option there are two stages, the first is to configure the RAID, and the second then is to select the RAID and perform a recovery, just as with a standard drive. ie a RAID will act logically in the same way as a single drive.

RAID recovery is often required when a RAID controller fails, or after an unsuccessful rebuild when a faulty drive is replaced. The CnW Recovery tools will assist in all cases where data is still possible to be recovered.

The basic RAID standards that are in common use are as below

RAID 0

It is often argued that RAID 0 is not actually a RAID as it has no redundancy. Data is striped between two or more disks. It is method so that 2 500GB disks can be made to logically look like a 1TB disk. With a suitable drive controller, both disks can be used at one time, and so can be faster than a single disk. Failure of a single disk means that 50% of data is lost. If a stripe is maybe 128KB long, then only some files, less than 128KB will be recoverable. If there is a partial failure of one disk, then CnW Recovery software will produce good results.

RAID 1

RAID 1 is 100% redundancy. The disk is imaged totally. If one disk fails, the data is still on the other. With a good hardware controller, performance will be the same as a normal disk. With a software controller, speed will suffer.

RAID 4

RAID 4 is the same as RAID 5 below, except the parity stripe is always on a fixed drive, and not a moving location. It is just as secure, but not always as fast. For reading, the RAID 5 routines will work, but parity will

be just one drive.

RAID 5

RAID 5 has at least 3 disks where one is a parity disk. This will accept a single disk failure without losing any data. Data is stored in stripes (maybe 128KB blocks at a time) and then the parity is calculated, and written on the final disk. It is common for the parity to be written on different disks for different stripes and so for a 4 disk array, the data may be organised as below

1	2	3	P
4	5	P	6
7	P	8	9
P	10	11	12

For reading, performance is good, but when writing a single sector it is necessary to also read the other unchanged sectors on the same stripe, and then update the parity sector. Thus what was a single write on a normal disk, becomes 2 reads and 2 writes. This can be done in software with a device driver, but for high performance, a hardware controller is required.

RAID 6

RAID 6 is similar to RAID 5 except there are two parity drives. It means that recovery is possible even if two drives fail. It can be used on 4 or more drives. The complex aspect is that the pattern used can be very varied. For this reason on RAID-6 a CnW option of 'Variations' can be specified - this is when there is not single pattern but several. (Only partially implemented in CnW)

JBOD

Job Bunch of Disks. This is the description given to multiple disks operating as a logical single drive. This may or may not include redundancy. One example is the HP media vault which has the string "Broadcom NAS Version 1.1 MBR Tag" as the start of disk 1. This is normally a Reiser FS disk stored on several sections of multiple disks. A specific mode of JBOD has been added in CnW to handle these disks.

RAID Recovery

There are a few approaches to RAID recovery. Depending on the application, and maybe forensic security, the disks may be read directly, or converted to a flat image to be processed in the same way as a single disk. The software is very flexible on how disks may be read, either as a physical disk, or as an image file - even an image stored over a network. A

completely failed disk can be marked as missing

Variations

Variations is an option where the RAID configuration pattern is longer than expected. For instance, with a RAID-6, 4 drive one would expect 3 lines of configuration, such as

```
1 2 P P
4 P P 3
P P 5 6
```

The possible problem with this configuration is that drive 1 and 4 could have twice the number of disk accesses than drives 2 and 3. Thus the actual pattern found has been for 24 stripes, rather than 6 stripes. To enter this with CnW it is necessary to set the variations value to 4, and fill in the data as follows (other RAID configurations may have different configurations).

```
1 2 P P
4 P P 3
P P 5 6

P 7 8 P
9 10 P P
12 P P 11

P P 13 14
P 15 16 P
17 18 P P

20 P P 19
P P 21 21
P 23 24 P
```

The variation value gives the number of variations - for most drives it will be 1, but the example above it was 4

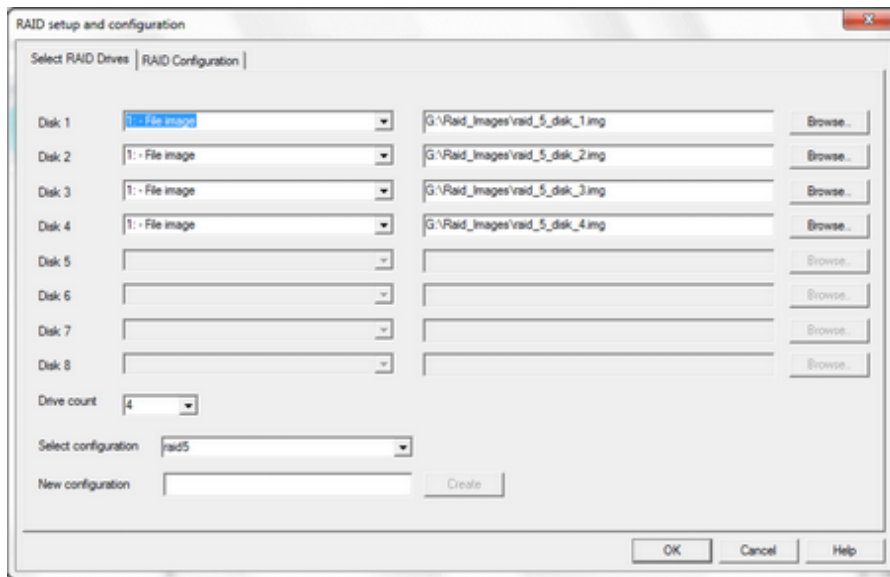
-o-

RAID drive selection

[Home](#)

There are three menus required to set up a RAID for CnW Recovery. The first is drive selection, and the second is the configuration, and third for JBODs

To create a new configuration, enter the required name in the New Configuration box, then select 'Create'. At this point select the number of drives, and then enter details for each drive.



To start configuring a RAID it is necessary to select the number of drives, and then define each drive. A drive can either be a

- Physical drive
- File image
- Defined as missing - only relevant for RAID-5.

The example above is for 4 disk images.

It is not essential to select the drives in the correct order as the analysis will determine this. However, if the order is known, it makes manual analysis rather easier

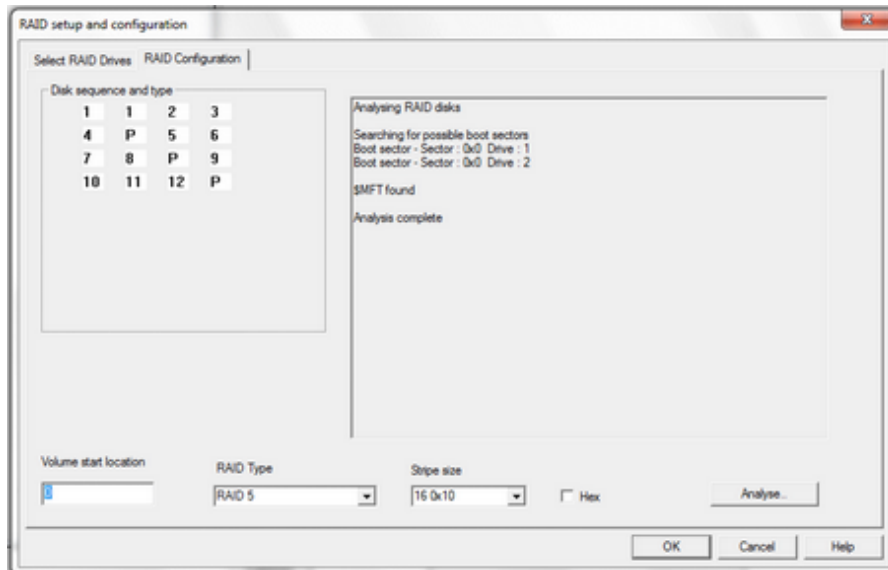
The RAID type is selected on this screen, and then the relevant configuration screen can be accessed to set up details of stripes etc.

-0-

RAID configuration

[Home](#)

The configuration of a RAID is very important. It can be done manually, but typically, CnW automatic analysis will assist



The areas that need to be configured with the RAID are

- RAID type
- Volume start location
- Stripe size
- Disk sequence and pattern used
- Stripe variations (normally 1)

Analyse

Some NAS drives start with both (all drives) mirrored, ie in RAID-1 configuration. The first part of the analysis is to detect this type of drive, and if found, it will automatically set the volume start location to the detected value. For instance, on a Lacie 2Big drive it will be set 0x1ea43d

The analyse function is designed to save a large amount of evaluation and testing. Once the drives have been selected, it goes through a series of tests which will determine the following parameters. These are currently as below

- RAID type, currently RAID 0 and RAID 5
- The stripe size which could be 0x10, 0x20, 0x40, 0x80, 0x100, 0x200, 0x400, 0x800 sectors

The process works in several stages (currently just for NTFS disks but other formats will be developed). The stages are as follows

- Find the MBR to establish where partition starts are, and type of partition
- Find BIOS Parameter Block (BPB) to establish location of MFT
- Find MFT entries on each disk to determine stripe size
- Analyse MFT entries to determine stripe order
- Save the new parameters with the RAID configuration

The analyse function will try and determine if the RAID is known to the software, in which case the parameters will be loaded automatically

Work out values by hand

Configuring a RAID is not easy. A good knowledge of file systems can help, along with being very happy reading Hex dumps. As a hint, the most useful type of file is a sequential file where the sequence is clear, and very often the best example of this is the \$MFT. A \$MFT may not be totally sequential but the MFT REF is a good number that increments with most MFT entries. By looking at this value one can very quickly determine the stripe size, and then try and determine the sequence between drives.

-0-

RAID boxes and configurations

[Home](#)

There are many RAID systems on the market. They can often be a small box with 2-8 drives in it, often used as a NAS (Network Attached Storage) device. Many of these systems actually contain a processor and a Linux operating system, which then controls a RAID 5 controller. To make the system slightly more complex for recovery, the disk may be in multiple partitions, so that Linux can be booted, and then a large partition where user data can be stored. From the host PC, the NAS just looks like a logical drive, that will store files. The host has no knowledge of how files are stored, which could actually be using any file system, though XFS does seem fairly common on new devices.

To determine the layout of a RAID is not trivial as it requires knowledge of file systems, and RAID structure. The following parameters have to be determined

- Stripe size
- Stripe configuration
- RAID start location.

Stripes

The stripe size is often the easiest to work out by hand. It is easiest to locate a long text file and it will then be clear when the text is not contiguous (allowing for file fragmentation). Typical stripe sizes are 0x80 and 0x100 sectors, though this can range from maybe 0x8 to 0x4000.

Configuration

The stripe configuration can be very difficult unless again there are some long, unfragmented text files. With a text file it is often easy to confirm that data is correct over a stripe boundary. Trial and error may be required and the following patterns may be seen for a 4 drive RAID 5

1 2 3 P	P 1 2 3	1 2 3 P
4 5 P 6	4 P 5 6	5 6 P 4
7 P 8 9	7 8 P 9	9 P 7 8
P 10 11 12	10 11 12 P	10 11 12 P

If the selected pattern is wrong, so will the data

RAID start location

Not all RAIDs are the same through the whole area of the disk. A common exception is to store the Linux operating systems files on all disks, ie as a RAID 1 configuration. The RAID 5 will then start at a location later on the

disk. To implement this feature, the RAID1 can be ignored, and the RAID can be set to start at a defined location.

Preconfigured RAID setups

To assist with reading proprietary RAIDs, certain setups will be distributed with the software. This will be if nothing else a basis for configuration, and any parameter may be changed.

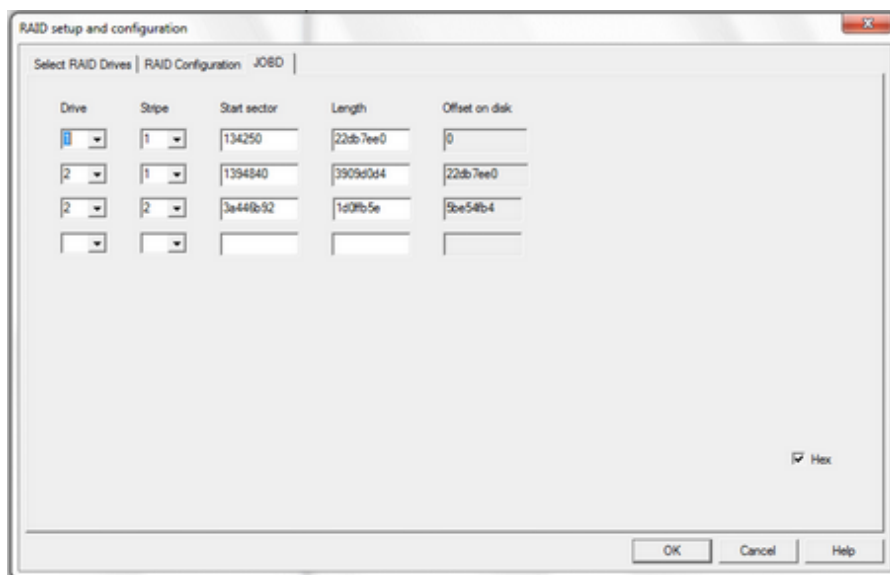
-0-

RAID JBOD

[Home](#)

Job Bunch of Disks are a series of disks configured so that they act as a single disk. Technically they are not a RAID because there is no redundancy, but of convenience, they are referred to as RAIDs.

CnW allows the user to select physical areas of the disk and then treat them as a logical device. The simplest configuration would be just appending the second (and third) disks to each previous one. Each disk can be set to have a logical stripe, and the length of the stripe. A stripe need not fill a disk, and there can be multiple stripes on a disk, not necessarily in sequence. The example shown below is based on a Broadcom NAS Version 1.1 set of disks



With a Broadcom disk the setup was as follows.

The parameters are set as above

When the drive was selected RAID, Sector 0 will display the string "BrcmSeMagicStr". Sector 0x80 will then show the Reiser SuperBlock with ReIsEr2Fs starting in byte 0x34. It may be necessary to set the partition to Reiser in the partitions option screen.

The Broadcom values are extracted for sector 0x1 (starting at 0x0)

```

00000000  00 00 00 00 00 00 00 0A - 00 00 00 00 00 02 00 00
00000010  00 00 00 00 00 02 00 0C - 00 00 00 00 00 02 00 00
00000020  00 00 00 00 00 13 42 50 - 00 00 00 00 22 DB 7E E0    BP    "Û~à
00000030  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

The dump above is for the first disk and the start sector in row 2 is 0x134250 with a length of 0x22DB7EE0. The first 2 start and length values are for control data, rather than user data

For disk two, there have been a few variations seen

```
00000000  00 00 00 00 00 0F 42 4C - 00 00 00 00 01 2A 05 F2      BL      * ò
00000010  00 00 00 00 3A 44 6B 92 - 00 00 00 00 1D 0F FB 5E      :Dk'    û ^
00000020  00 00 00 00 01 39 48 40 - 00 00 00 00 39 09 D0 D4      9H@     9 ðÔ
```

The data above apparently has three sections, but from experimentation the first one was not used. On another disk seen, the second disk had only a single section of data, with it's data in the first 16 bytes of sector 2

Volume Start

The volume start is normally set to zero but on some RAID system, the RAID configuration starts at a certain sector. Typically, all sectors before this point are RAID-1, ie mirrored between drives.

Concatenate drives

This option is a quick way to set up drives that are appended to each other. It assumes that all of the first drive is accessed, followed by the second drive, and maybe third drive. In these cases the stripe value is always 1.

Analyse Media vault..

The analyse feature is currently for HP Media Vault disks. It will try and configure the disks and section starts based on the meta data stored on the drives

-0-

Typical RAID setup parameters

[Home](#)

There are many off the shelf RAID systems on the market. CnW will handle a lot of systems, but sometimes working out the parameters can be complex.

The list below is from actual RAID systems. These may not be typical values but can be used as a starting point for working out the drive setup.

For many configurations, the start of the drive is actually RAID 1 (ie all disks have the same information) and then the data section is in RAID 0, RAID 5 etc.

Internal Apple RAID 0 with 3 drives

Stripe size 64 sectors (32K)
RAID 0, disk order 2,3,1
Volume start location 0x64028
File system, HFS+

Lacie 2Big NAS with 2 drives, RAID 0

Stripe size 128 (64K)
RAID 0 disk order 1,2
Volume start location 0x1ea43d
File system XFS

-0-

HP Mediavault recovery

[Home](#)

The HP Mediavault is a fairly common RAID system. It is more common for the RAID controller to fail than the disks. Thus the user ends up with a pair of good disks, but no means to read them.

With CnW RAID option, it is often possible to recover all of the files from the Reiser FS disk drives. To enable the RAID option on the demo program, please contact CnW at info@cnwrecovery.com

How to recognise the disks?

HP Media vault disks have a non standard boot sector, ie sector 0. A typical sector is boot sector is shown below.

```

00000000  42 72 6F 61 64 63 6F 6D - 20 4E 41 53 20 56 65 72  Broadcom NAS Ver
00000010  73 69 6F 6E 20 31 2E 31 - 20 4D 42 52 20 54 61 67  sion 1.1 MBR Tag
00000020  00 00 00 10 18 01 10 31 - 53 59 53 54 45 4D 00 00  1SYSTEM
00000030  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
00000040  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
00000050  00 00 00 00 DA C6 87 74 - 21 EB 1C D0 D8 E0 A6 74  ÚË+!èÐà|t
00000060  95 58 19 0D 00 00 00 00 - F2 00 00 00 00 00 00 00  •X  ò

```

The important part of the boot sector is the string 'Broadcom NAS Version 1.1 MBR tag'. However, it is quite common for boot sectors to be overwritten by well meaning people trying to create a valid boot sector. Thus sector 1 can also be examined and is quite distinctive. Examples of sector 1 are shown below

Disk 1

```

00000000  00 00 00 00 00 00 00 0A - 00 00 00 00 00 02 00 00
00000010  00 00 00 00 00 02 00 0C - 00 00 00 00 00 02 00 00
00000020  00 00 00 00 00 13 42 50 - 00 00 00 00 22 DB 7E E0  BP  "Û~à
00000030  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

Disk 2

```

00000000  00 00 00 00 00 0F 42 4C - 00 00 00 00 01 2A 05 F2  BL  * ò
00000010  00 00 00 00 3A 44 6B 92 - 00 00 00 00 1D 0F FB 5E  :Dk'  û^
00000020  00 00 00 00 01 39 48 40 - 00 00 00 00 39 09 D0 D4  9Hè  9 ÐÔ

```

The sectors above have two pieces of information, the start of a section, and the length. The numbers are big endian, 64 bit numbers. Disk 1 typically starts with 2 control sections, followed by data section(s).

Configurations

The Broadcom is meant to come in various configurations, including RAID 1 and JBOD. For RAID 1, both disks should be identical and so can be read without the RAID option. For RAID 1, the total capacity will be the same as a single drive. Both drives will need to be the same capacity

JBOD is when disks are added, maybe separately and the total capacity is

nearly the same as the two drives added together. Drives can be different capacities. To read this type of drive, RAID JBOD option is required.

The final configuration is as two separate drives. For this the RAID option is not required and CnW will read each disk as a Reiser drive

To recovery HP MediaVault JBOD

The first stage is to create a new RAID configuration and give it a name (such as Broadcom, or HPVault). Enter the drives being used - or drive images, along with the number of drives. Also set the configuration to be JBOD

Select the JBOD configuration screen and select Analyse. This should fill in the location boxes and lengths.

The configuration is now complete, so select 2: - RAID as the drive type

Select Recover, and if all OK, the Resiser FS format should be seen, and the Unix options screen will be displayed. Select the location for the data to be recovered to, and press OK.

-0-

Fragmented files

[Home](#)

Most file systems will at some time or another create files that are not sequential. The best file systems work hard to prevent or reduce the number of fragmented files, but ultimately, with a fairly full disk, and a large file, there are no longer contiguous areas to save the file. The other common reason for fragmented files are files that grow. These growing files could be logs, or maybe e-mail systems which expand every day.

Symptoms of fragmented files

If a file has been fragmented, then the start will look OK. ie The signature will be valid, however it is unlikely that the end of the file will be in the correct location. If a fragmented photo is viewed, then it will almost certainly be incomplete. Often the bottom of the picture will be blank, or possibly view parts of different pictures. For a video, most will not play unless the complete video and control data is present.

CnW has several file validate routines to check the integrity of files and determine if defragmenting is relevant.

How to process fragmented files

When doing a recovery using a logical file system such as NTFS, XFS all fragmentation is taken care of by the file system handler. For deleted NTFS, it is also normal for the fragmentation information to be retained. The problem comes with deleted FAT disks and those disk that have no file system information left. FAT is not used much on PC hard drives these days, but is very common on memory chips, camera and videos and also often on external USB drives.

The links below indicate how each type of fragmented file should be processed using CnW Recovery and data carving.

[AVCHD - high resolution video](#)

[AVI video](#)

[3GP, F4YP, MOV, MP4 video](#)

[JPEG Photos](#)

[Zip, DOCX, XLXS, ODT files](#)

[White paper on topic](#)

Non video files

CnW will process fragmented JPEGs. The results are best from a camera

memory device as hard drives tend to be rather large and JPEGs can get scattered a lot.

CnW has routines for ZIP and Word but they have not been developed to a very high level,

-o-

Fragmented 3GP/MP4 files reconstruction

[Home](#)

3GP/MP4 Files

3GP/MP4 are all part of the Quick Time file structure. A Quick Time file has three main elements

- ftyp - the file header
- moov - file meta data
- mdat - the file video and audio data

Logically, the main file can be either of the two sequences below

ftyp-mdat-moov or ftyp-moov-mdat

The format is very common with current disk or memory chip video recorders (rather than min dvd recorders). Most recorders use FAT32 as the file system which means when deleted, the files may be fragmented.

Why are 3GP/MP4 files fragmented?

The two main reasons why the files are fragmented either because of general FAT32 fragmentation, or due to the way they have been recorded. When an individual file, or group of files are deleted on FAT32 there is an area of unallocated data space. New files will be written to this space but if larger than the first gap, the file will be split into two or more fragments. The chaining is controlled by the FAT (file allocation table) and so invisible to the user. When files are deleted, so is the relevant allocation information in the FAT, and so recovery from data carving is not immediately possible.

The second reason for fragmentation is rather more obscure. When a video is recorded the mdat segment takes most space, but is continuously indexed by the moov segment. Until recording is finished, the size of the

moov and mdat segments is unknown. The mdat segment is large and so can not be cached in camera memory and must be written directly to the disk. The moov segment contains many atoms that have pointers to the mdat, and these pointers are often absolute values from the start of the main file. Different camera manufacturers have devised different approaches to the problem. These are outlined below and automatically detected by CnW Recovery software to reconstruct the fragmented files

mdat-ftype-moov

In this format the camera records the mdat straight to the disk, and when finalised the header and moov segments are added to the next cluster after the mdat. By manipulating the FAT, logically the file will look like ftyp-mov-mdat. If a standard program tries data carving, then the incorrect ftyp-moov will be applied to the following mdat and the video will not play. CnW data carving and fragment processing takes care of this issue.

ftype - preallocated moov - mdat - trak

In the above approach an attempt to pre-allocate the moov area is used. The moov area is allocated an area at the start of the file and the large atoms, such as 'stsz' all start on cluster boundaries. These atoms are then padded with a 'free' so that they remain cluster sized. For some reason, on one example seen, the audio 'trak' atom is stored after the full mdat segment.

General defragmentation recovery

The examples above show fragmentation in a known pattern. General defragmentation shows no such patterns and processing is more complex. It falls into two main stages with the assumption that all data is present.

Recovery when there is no moov segment

If the camera is switched off before recording has finished, or maybe dropped or battery removed then cases can exist where there is a mdat segment but no moov segment. CnW has a solution for this that will work with certain file types (but currently not all). The technique used is to use a known good file and extract moov fragment as a template. This will give fixed values for the camera and then the variables, eg stsz tables are reconstructed.

This technique does require knowledge of each type of codec used. For this reason, not every MP4 can currently be recovered this way, but the list is growing. CnW are happy to add support for anyone who has a failed file,

and can provide complete file from the same camera.

-0-

Typical 3GP corruptions

[Home](#)

3GP files can be corrupted for several reasons. This page will describe some of the common reasons, and how they can be reconstructed.

Reconstruction after file deletion

Unfinalised

Video camera files are not typically found unfinalised, but it could happen if there was a camera failure, or maybe the memory chip or battery was removed before completion. The most likely indication of this would be 'mdat' segment length is zero. This means that it is very likely that the 'moov' segment has not been written.

The solution to the above is to create a new moov segment. As the possible variations are very large, the approach does require a sample valid file, from which the moov parameters can be analysed and used. CnW software will automatically scan the media for a suitable file and use it. The approach does require the locations of each frame to be found by parsing the mdat segment. Frames are indicated in several ways depending on which codec has been used to record the video. The codec will be determined from the sample moov segment in the sample file.

-0-

Fragmented Zip and DOCX files

[Home](#)

Zip files are used for both general archives, preparing to send multiple files but also as storage for current word processing packages. The current packages are Office, 2007 and later, and Open Office, .ODT files.

The reason for the zip framework is that a .DOCX files is based on several XML files that are very verbose, and easily compressed. Thus a file that would be maybe 100K is reduced to nearer 10K. Thus zipping the files saves space, and also reduces the chance that file will be fragmented. If fragmented, the CnW data carving function will recover many such files, and the process is described briefly below

Zip files are fairly straight forward to defragment as they have a well defined data structure, helped by sections with pointers and lengths. As mentioned about, many DOCX files are fairly small, and will not be fragmented more than a few times at worst case - the exception is when they have embedded photos.

The Zip file structure.

The basic file structure is well documented, (<http://www.phpconcept.net/pclzip/pkzip.txt> is one such link) so the following is just a brief outline.

File signature

The basic signature is 'PK' followed by 0x03 0x04 which is a local file header

```
00000000  50 4B 03 04 14 00 00 00 - 08 00 47 72 23 39 CC 1E  PK      Gr#9İ
00000010  5C F4 57 3B 00 00 68 A0 - 00 00 18 00 00 00 6C 69  \ôW;  h    li
00000020  62 2F 61 75 74 6F 2F 57 - 69 6E 33 32 2F 57 69 6E  b/auto/Win32/Win
00000030  33 32 2E 64 6C 6C ED 7D - 0B 78 54 D5 D5 E8 C9 7B  32.dlli} xTÔÔëË{
00000040  80 81 09 90 60 84 00 03 - 24 10 CA 6B F2 22 C9 3C  €••„ $Ëkô"Ë<
```

0x00 local file header signature	4 bytes	(PK 0x03 0x04)
0x04 version needed to extract	2 bytes	
0x06 general purpose bit flag	2 bytes	
0x08 compression method	2 bytes	
0x0a last mod file time	2 bytes	
0x0c last mod file date	2 bytes	
0x0e crc-32	4 bytes	
0x12 compressed size	4 bytes	
0x16 uncompressed size	4 bytes	
0x1a filename length	2 bytes	
0x1c extra field length	2 bytes	

filename (variable size)
extra field (variable size)

The example above shows that the compressed size of the file is 0x3b57 and uncompressed is 0xa068. The file name is 0x18 bytes long and so the compressed string starts at location 0x1e (length of header) + 0x18 (name length), ie offset 0x36. As we know that this section is 0x3b57 bytes long, the next PK header will be at location 0x3b57 + 0x36, ie 0x3b8d.

On a fragmented file, the technique is to search for a PK header which has an offset within a cluster of 0x3b8d. Thus for a cluster size of 0x4000 bytes, it would be offset 0x3b8d, but for a cluster size of 0x1000 bytes, the offset would be 0xb8d. With a limited number of Zip files, the chance of a miss match is limited. The header sumcheck can be verified to make sure it is valid PK header.

```

00003B80  16 5B 24 20 FF 62 A5 B5 - 76 AB F0 3F 00 50 4B 03  [$ ŷbŷµv«ð? PK
00003B90  04 14 00 00 00 08 00 DC - 6A 23 39 E2 F1 1B 49 02  Űj#9āñI
00003BA0  3B 00 00 61 A0 00 00 1C - 00 00 00 6C 69 62 2F 61  ; a lib/a
00003BB0  75 74 6F 2F 57 69 6E 33 - 32 2F 57 69 6E 33 32 2E  uto/Win32/Win32.
00003BC0  64 6C 6C 2E 41 41 41 ED - 7D 7B 78 54 D5 B5 F8 C9  dll.AAAi}{xTÔµøĒ

```

As can be seen a new PK header is found in the correct location. This process can be continued through the file.

Central Register

Towards the end of the file a central register is stored. This is a directory of all files within the Zip file

```

000033E0  2C 5B 02 17 72 AC F2 FF - 01 00 00 FF FF 03 00
50      , [ r-òŷ ŷŷ P
000033F0  4B 01 02 2D 00 0A 00 00 - 00 00 00 00 00 21 00
5E      K - ! ^
00003400  C6 32 0C 27 00 00 00 27 - 00 00 00 08 00 00 00
00      Æ2 ' '
00003410  00 00 00 00 00 00 00 00 - 00 00 00 00 00 6D 69
6D      mim
00003420  65 74 79 70 65 50 4B 01 - 02 2D 00 14 00 06 00
08      etypePK -

```

0x00	central file header signature	4 bytes	PK 0x01 0x02
0x04	version made by	2 bytes	
0x06	version needed to extract	2 bytes	
0x08	general purpose bit flag	2 bytes	
0x0a	compression method	2 bytes	
0x0c	last mod file time	2 bytes	
0x0e	last mod file date	2 bytes	
0x10	crc-32	4 bytes	
0x14	compressed size	4 bytes	
0x18	uncompressed size	4 bytes	
0x1c	filename length	2 bytes	
0x1e	extra field length	2 bytes	

0x20 file comment length	2 bytes
0x22 disk number start	2 bytes
0x24 internal file attributes	2 bytes
0x26 external file attributes	4 bytes
0x2a relative offset of local header	4 bytes

0x2e filename (variable size)
extra field (variable size)
file comment (variable size)

The central regisiter can be used to verify the file structure and that all elements are present and correct. If there is an error, then it is likely that somewhere there has been a false positive match.

Final header

The final header is basically a pointer to the start of the central regisiter

end of central dir signature	4 bytes	(PK 0x05 0x06)
number of this disk	2 bytes	
number of the disk with the start of the central directory	2 bytes	
total number of entries in the central dir on this disk	2 bytes	
total number of entries in the central dir	2 bytes	
size of the central directory	4 bytes	
offset of start of central directory with respect to the starting disk number	4 bytes	
.ZIP file comment length	2 bytes	
.ZIP file comment	(variable size)	

```
00003540    74 79 6C 65 73 2E 78 6D - 6C 50 4B 05 06 00 00
00      types.xmlPK
00003550    00 06 00 06 00 5A 01 00 - 00 EF 33 00 00 00 00
          Z   i3
```

CnW Zip Recovery

The CnW routine can be called after the data carving has detected corrupted - possibly fragmented - Zip files. It will run the above techniques to scan the hard drive / memory chip for fragments that fit the zip file.

-0-

Recognising Sectors

[Home](#)

An important part of data recovery is being able to recognise important system sectors. For an experienced investigator this becomes second nature, but for anyone starting with data recovery it can be rather daunting. This section gives sample dumps of critical sectors and indicates where there will be found, and their function.

With the exception of the Master Boot Record each sector type is specific to an operating system, although there can be similarities

[Master Boot Record](#)

[GUID Sectors](#) - as on many Macintosh systems

[BIOS Parameter Block](#)

[FAT directory entry](#)

[NTFS Directory entry, MFT](#)

[Disk clusters](#) - how to work out their size

[VMFS sectors](#)

-0-

Master Boot Record

[Home](#)

The sector below is a typical master boot record, ie sector 0 of a disk

```

000000  33 C0 8E D0 BC 00 7C FB - 50 07 50 1F FC BE 1B 7C  3ÄŽD¼ |ûPPü¼|
000010  BF 1B 06 50 57 B9 E5 01 - F3 A4 CB BE BE 07 B1 04  ¼ PW¹ääó¼E¼¼±
000020  38 2C 7C 09 75 15 83 C6 - 10 E2 F5 CD 18 8B 14 8B  8,| ufEäöÍ<<
000030  EE 83 C6 10 49 74 16 38 - 2C 74 F6 BE 10 07 4E AC  ifEIt8,tö¼N¬
000040  3C 00 74 FA BB 07 00 B4 - 0E CD 10 EB F2 89 46 25  < tú» ´Íëð¼F¼
000050  96 8A 46 04 B4 06 3C 0E - 74 11 B4 0B 3C 0C 74 05  -ŠF´<t´<t
000060  3A C4 75 2B 40 C6 46 25 - 06 75 24 BB AA 55 50 B4  :Äu+@EF%u$»ªUP´
000070  41 CD 13 58 72 16 81 FB - 55 AA 75 10 F6 C1 01 74  AÍXr¼UªuöÄt
000080  0B 8A E0 88 56 24 C7 06 - A1 06 EB 1E 88 66 04 BF  Šà^vŞÇ;ë^fç
000090  0A 00 B8 01 02 8B DC 33 - C9 83 FF 05 7F 03 8B 4E  ,<Ü3Éfÿ¼<N
0000A0  25 03 4E 02 CD 13 72 29 - BE 46 07 81 3E FE 7D 55  %NÍr)¼F¼p}U
0000B0  AA 74 5A 83 EF 05 7F DA - 85 F6 75 83 BE 27 07 EB  ºtZfi¼Ü...öuf¼'ë
0000C0  8A 98 91 52 99 03 46 08 - 13 56 0A E8 12 00 5A EB  Š~'R¼F V è Zë
0000D0  D5 4F 74 E4 33 C0 CD 13 - EB B8 00 00 00 00 00 00  ÖOtä3ÄÍë,
0000E0  56 33 F6 56 56 52 50 06 - 53 51 BE 10 00 56 8B F4  V3öVVVRPSQ¼ V<ö
0000F0  50 52 B8 00 42 8A 56 24 - CD 13 5A 58 8D 64 10 72  PR, BŠV$ÍZXdr
000100  0A 40 75 01 42 80 C7 02 - E2 F7 F8 5E C3 EB 74 49  @uBEÇä÷ø^ÄëtI
000110  6E 76 61 6C 69 64 20 70 - 61 72 74 69 74 69 6F 6E  nvalid partition
000120  20 74 61 62 6C 65 00 45 - 72 72 6F 72 20 6C 6F 61  table Error loa
000130  64 69 6E 67 20 6F 70 65 - 72 61 74 69 6E 67 20 73  ding operating s
000140  79 73 74 65 6D 00 4D 69 - 73 73 69 6E 67 20 6F 70  ystem Missing op
000150  65 72 61 74 69 6E 67 20 - 73 79 73 74 65 6D 00 00  erating system
000160  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

```

000170  00 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000180  00 00 00 8B FC 1E 57 8B - F5 CB 00 00 00 00 00 00    < üW< öE
000190  00 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001A0  00 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001B0  00 00 00 00 00 00 00 00 00 - CA EE BA 36 00 00 00 01    Ei°6
0001C0  01 00 07 FE 7F D7 3F 00 - 00 00 99 FF 14 13 00 00    b*x?  mÿ
0001D0  00 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001E0  00 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001F0  00 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 55 AA    Uª

```

The boot sector has three main sections, described as follows

The first section is optional, and includes all bytes upto 0x1BE. This is code that is used for a disk to boot from and is blank for non bootable disks, such as camera memory chips. The boot data can be different for each machine, but typically it does start with the same sequence, such as 0x33 0xc0. It is also typical, such as above to have some text as possible warning messages

The final two bytes of the sector must be 0x55 0xAA This is the same as several other operating system control sectors

The most important area of the boot sector is the partition map starting at location 0x1BE. There are infact 4 possible tables, each of 16 bytes in length. See [Partition Table Structure](#) for full details.

The points to look for to recognise the block is that the fact they always end with 0x55 0xAA and there are 1 - 4 partition records starting at 0x1BE

-0-

GUID Partition sectors

[Home](#)

GUID Master Boot records are standard - as in [Master Boot Record](#), but the partition table entry is rather different. The partition type, as described in byte 4 is set to 0xEE and the partition start, bytes 8-11 are normally set at 1. Sector 1 then has the complete partition information.

Sector 0

```

000000  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000010  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000020  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000030  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000040  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000050  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000060  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000070  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000080  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000090  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000A0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000B0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000C0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000D0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000E0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0000F0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000100  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000110  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000120  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000130  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000140  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000150  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000160  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000170  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000180  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000190  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001A0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001B0  00 00 00 00 00 00 00 00 - 8C CC 51 6B 00 00 00 FE
0001C0  FF FF EE FE FF FF 01 00 - 00 00 6F 59 1C 1D 00 00
0001D0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001E0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001F0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 55 AA

```

EFI PART \
 æĒ' "
 oY " "
 NY Nñ*•ze-I
 †İĖŮ•...
 € € μ×æ<
 U^a

Sector 1 Partition table header

```

000000  45 46 49 20 50 41 52 54 - 00 00 01 00 5C 00 00 00
000010  E6 CB B4 84 00 00 00 00 - 01 00 00 00 00 00 00 00
000020  6F 59 1C 1D 00 00 00 00 - 22 00 00 00 00 00 00 00
000030  4E 59 1C 1D 00 00 00 00 - 4E F1 2A 90 7A 65 97 49
000040  86 05 CF CA DB 95 81 85 - 02 00 00 00 00 00 00 00
000050  80 00 00 00 80 00 00 00 - B5 D7 E6 8B 00 00 00 00
000060  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

EFI PART \
 æĒ' "
 oY " "
 NY Nñ*•ze-I
 †İĖŮ•...
 € € μ×æ<

The rest of the sector is all zeros

```

0001E0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001F0  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

The sector always starts with the string EFI PART followed by the version number (1) and record length (0x5c)

Offset 0x18 give the location of this sector (1)

Offset 0x20 is the offset of the spare EFI header (0x1d1c59ef)

Offset 0x28 is the start of the data area of the disk (0x22) normally just after the partition entries

Offset 0x48 start of partition entries, normally 2

Sector 2-33, Partition entries

```

00000000 16 E3 C9 E3 5C 0B B8 4D - 81 7D F9 2D F0 02 15 AE ăÉă\ ,M•}ù-ø @
00000010 6E D9 3E F6 CE 0D B5 42 - A5 B8 8A 0B 10 6A 55 E3 nŮ>ôî µBŸ,Š jUă
00000020 22 00 00 00 00 00 00 00 - 21 00 04 00 00 00 00 00 " !
00000030 00 00 00 00 00 00 00 00 - 4D 00 69 00 63 00 72 00 M i c r
00000040 6F 00 73 00 6F 00 66 00 - 74 00 20 00 72 00 65 00 o s o f t r e
00000050 73 00 65 00 72 00 76 00 - 65 00 64 00 20 00 70 00 s e r v e d p
00000060 61 00 72 00 74 00 69 00 - 74 00 69 00 6F 00 6E 00 a r t i t i o n
00000070 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
00000080 A2 A0 D0 EB E5 B9 33 44 - 87 C0 68 B6 B7 26 99 C7 ċ Đěăî3D†Àhŧ ·&™Ç
00000090 0A CD 32 5D 7E A5 35 48 - 9F B3 D7 0A A5 1F 77 EF í2]~Ÿ5HŸ³× Ÿwî
000000A0 00 08 04 00 00 00 00 00 - FF 2F 51 5D 01 00 00 00 Ÿ/Q]
000000B0 00 00 00 00 00 00 00 00 - 42 00 61 00 73 00 69 00 B a s i
000000C0 63 00 20 00 64 00 61 00 - 74 00 61 00 20 00 70 00 c d a t a p
000000D0 61 00 72 00 74 00 69 00 - 74 00 69 00 6F 00 6E 00 a r t i t i o n
000000E0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000000F0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
00000100 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

EFI stands for Extensible Firmware Interface. EFI is designed to improve upon the existing Partition table design, which in particular has a 32 bit limit on sector numbers. 32 bit addressing will allow for 2TB of disk. Although this is not a problem at the moment, 1TB disks are becoming common, and 2TB are just being announced. The address fields are now 64 bit rather than 32 bit. This will give a few years grace in capacity.

Sector 2, and following sectors describe each partition type. Rather than a single byte, a GUID is used. Each partition is described by 0x80 (128) byte record, and so in the example above, there are two partitions. The first one

```
16 E3 C9 E3 5C 0B B8 4D - 81 7D F9 2D F0 02 15 AE
```

is a reserved microsoft partition. It starts at sector 0x22 and has a length of 0x40021

The second partition is Microsoft Data, and can be a normal NTFS partition. The GUID is

```
A2 A0 D0 EB E5 B9 33 44 - 87 C0 68 B6 B7 26 99 C7
```

and the starting sector is 0x40800. The length is 0x15d512fff sectors, or about 2.9TB. It shows that this value is greater than 32 bits, and hence the requirement for the EFI partition data.

A GUID is a Globally Unique ID. These are numbers which should be unique. A series of such numbers have been defined for different types of disk partition, eg Microsoft Data Partition, Apple HFS+. There are also defined numbers for Linux, Solaris, HP-UX. CnW currently recognise a few of these, but the list will grow. The second 16 bytes of partition entry is a GUID for the specific drive. This can be treated as a unique partition serial number.

BIOS Parameter Block BPB

[Home](#)

The partition boot sector contains the BIOS Parameter block is used to define the details of a logical partition. The sector is pointed to from the entry in the partition table. A very common location for the first BPB on a disk is sector 0x3F (63). For camera memory chips, sector 0x20 (32) is also common

```

000000 EB 52 90 4E 54 46 53 20 - 20 20 20 00 02 08 00 00   èR•NTFS
000010 00 00 00 00 00 00 F8 00 00 - 3F 00 FF 00 3F 00 00 00   ø ? ý ?
000020 00 00 00 00 00 00 80 00 00 - 98 FF 14 13 00 00 00 00   e e `ý
000030 00 00 0C 00 00 00 00 00 00 - F9 4F 31 01 00 00 00 00   ù01
000040 F6 00 00 00 01 00 00 00 00 - 12 73 96 1C A6 96 1C 24   ö s-!-$
000050 00 00 00 00 FA 33 C0 8E - D0 BC 00 7C FB B8 C0 07   ú3ÄŽĐ¼ |û,À
000060 8E D8 E8 16 00 B8 00 0D - 8E C0 33 DB C6 06 0E 00   Ž0è , ŽÄ3ÜÈ
000070 10 E8 53 00 68 00 0D 68 - 6A 02 CB 8A 16 24 00 B4   ès h`hjÈŠŠ`
000080 08 CD 13 73 05 B9 FF FF - 8A F1 66 0F B6 C6 40 66   ís`ýŸŠñfŸ@ef
000090 0F B6 D1 80 E2 3F F7 E2 - 86 CD C0 ED 06 41 66 0F   ŸÑeâ?÷â†íÄíAf
0000A0 B7 C9 66 F7 E1 66 A3 20 - 00 C3 B4 41 BB AA 55 8A   ·Éf÷áfÉ Ä`A»ªUŠ
0000B0 16 24 00 CD 13 72 0F 81 - FB 55 AA 75 09 F6 C1 01   $ ír·ûUªu öÁ
0000C0 74 04 FE 06 14 00 C3 66 - 60 1E 06 66 A1 10 00 66   tþ Äf`f; f
0000D0 03 06 1C 00 66 3B 06 20 - 00 0F 82 3A 00 1E 66 6A   f; ,: fj
0000E0 00 66 50 06 53 66 68 10 - 00 01 00 80 3E 14 00 00   fPSfh e>
0000F0 0F 85 0C 00 E8 B3 FF 80 - 3E 14 00 00 0F 84 61 00   ... è³ý€> „a
000100 B4 42 8A 16 24 00 16 1F - 8B F4 CD 13 66 58 5B 07   ´BŠ$ <ôífx[
000110 66 58 66 58 1F EB 2D 66 - 33 D2 66 0F B7 0E 18 00   fXfXe-f3Öf·
000120 66 F7 F1 FE C2 8A CA 66 - 8B D0 66 C1 EA 10 F7 36   f÷ñpÄŠÈf<ĐfÄê÷6
000130 1A 00 86 D6 8A 16 24 00 - 8A E8 C0 E4 06 0A CC B8   †ÖŠ$ ŠeÄÄ î,
000140 01 02 CD 13 0F 82 19 00 - 8C C0 05 20 00 8E C0 66   í , ¢Ä ŽÄf
000150 FF 06 10 00 FF 0E 0E 00 - 0F 85 6F FF 07 1F 66 61   ý ý ...oýfa
000160 C3 A0 F8 01 E8 09 00 A0 - FB 01 E8 03 00 FB EB FE   Ä øè ûè ûèþ
000170 B4 01 8B F0 AC 3C 00 74 - 09 B4 0E BB 07 00 CD 10   ´<ð-< t´» í
000180 EB F2 C3 0D 0A 41 20 64 - 69 73 6B 20 72 65 61 64   èòÄ A disk read
000190 20 65 72 72 6F 72 20 6F - 63 63 75 72 72 65 64 00   error occurred
0001A0 0D 0A 4E 54 4C 44 52 20 - 69 73 20 6D 69 73 73 69   NTLDR is missi
0001B0 6E 67 00 0D 0A 4E 54 4C - 44 52 20 69 73 20 63 6F   ng NTLDR is co
0001C0 6D 70 72 65 73 73 65 64 - 00 0D 0A 50 72 65 73 73   mpressed Press
0001D0 20 43 74 72 6C 2B 41 6C - 74 2B 44 65 6C 20 74 6F   Ctrl+Alt+Del to
0001E0 20 72 65 73 74 61 72 74 - 0D 0A 00 00 00 00 00 00   restart
0001F0 00 00 00 00 00 00 00 00 - 83 A0 B3 C9 00 00 55 AA   f ³É Uª

```

Another example

```

000000 EB 3C 90 4D 53 44 4F 53 - 35 2E 30 00 02 01 02 00   è<•MSDOS5.0
000010 02 00 02 60 F4 F8 F3 00 - 3F 00 FF 00 20 00 00 00   `ðøó ? ý
000020 00 00 00 00 00 00 29 B7 - E1 51 58 4E 4F 20 4E 41   )·áQXNO NA
000030 4D 45 20 20 20 20 46 41 - 54 31 36 20 20 20 33 C9   ME FAT16 3É

```

Two easy points to help recognise the sector are the terminating 0x55 0xAA characters, but also the operating system name starting at byte 3. In the examples above these are NTFS and MSDOS5.0

For FAT disks, you also expect to see FAT12 or FAT16 or FAT32 near the top of the sector

The first bytes of the sector are a JMP instruction so normally starts 0xEB

Logically, the FDC data starts at byte 0xB. Full details can be found in [FDC Descriptor for FAT](#) or [FDC Descriptor for NTFS](#)

NTFS details

Bytes 0x0B-0x0C	00 02	0x200 or 512 bytes per sector
Byte 0x0D	08	8 sector per cluster (normal value). Possible values are 1,2,4,8,16,32,64
Bytes 0x0E-0x0F	00 00	2 reserved sectors NTFS always starts at 0.
Byte 0x10-0x12	00 00 00	Always zero
Bytes 0x13-0x14	00 00	Always zero
Byte 0x15	F8	Media type - always F8 for hard drive
Bytes 0x16-0x17	00 00	Always zero
Bytes 0x18-0x19	3F 00	Not checked by NTFS
Bytes 0x1A-0x1B	FF 00	Not checked by NTFS
Bytes 0x1C-0x1F	3F 00 00 00	Not checked by NTFS
Bytes 0x20-0x23	00 00 00 00	Must be
Bytes 0x24-0x27	80 00 80 00	Not checked by NTFS
Bytes 0x28-0x2F	98 FF 14 13 00 00 00 00	Total sectors on hard drive
Bytes 0x30-0x37	00 00 0C 00 00 00 00 00	Logical cluster number for \$MFT
Bytes 0x38-0x3F	F9 4F 31 01 00 00 00 00	Logical cluster number for \$MFTMirr
Byte 0x40	F6	Cluster per MFT record
Bytes 0x41-0x43	00 00 00	Not used by NTFS
Bytes 0x44	01	Clusters per Index Buffer
Bytes 0x45-0x47	00 00 00	Not used by NTFS
Bytes 0x48-0x4F	12 73 96 1C A6 96 1C 24	Volume serial number

Bytes 0x50-0x53 00 00 00 00 Not used by NTFS

-0-

FAT directory entry

[Home](#)

A FAT file system has two basic types of directory, the root directory and subdirectories. On FAT12 and FAT16, the root directory is in a fixed location, just after the FAT, and it is a fixed size. On FAT32, the start of the root directory is defined in the BPB, and the directory can be any length.

A directory is made up of 0x20 byte records. The only difference between a root and subdirectory, is that the sub directory always starts with two records showing the current location, and the parent directory location. These entries are ".", ".." and "..". The file attribute for both is 0x10 showing it is a subdirectory. Typically, the first entry for a root directory is the volume label, with a file attribute of 0x08

Root directory

000000	50 4F 57 45 52 53 48 4F - 54 20 20 08 00 00 00 00	POWERSHOT
000010	00 00 00 00 00 00 C7 09 - 49 37 00 00 00 00 00 00	Ç I7
000020	E5 4D 47 5F 31 36 39 36 - 4A 50 47 20 00 1B 8B 86	âMG_1696JPG <†
000030	49 37 4C 37 00 00 4A B5 - 46 37 02 00 74 82 2C 00	I7L7 JpF7 t,,
000040	E5 4D 47 5F 31 36 32 36 - 4A 50 47 20 00 89 8D 86	âMG_1626JPG %†
000050	49 37 49 37 00 00 0B 01 - 43 37 44 16 18 C7 43 00	I7I7 C7D ÇC
000060	E5 54 4C 41 4E 54 41 20 - 4A 50 47 20 18 2E 6B 97	âTLANTA JPG .k-
000070	49 37 4D 37 00 00 05 0E - 3E 35 28 38 75 11 04 00	I7M7 >5(8u
000080	E5 45 41 43 48 20 20 20 - 4A 50 47 20 18 70 6B 97	âEACH JPG pk-
000090	49 37 4A 37 00 00 F4 08 - 43 35 31 3A F3 FC 01 00	I7J7 ôC51:ôü
0000A0	E5 45 41 43 48 5F 32 20 - 4A 50 47 20 18 90 6B 97	âEACH_2 JPG *k-
0000B0	49 37 49 37 00 00 D7 6A - 44 35 30 3B 0B 26 03 00	I7I7 ~xjD50; &
0000C0	E5 45 41 43 48 4E 43 20 - 4A 50 47 20 18 0A 6C 97	âEACHNC JPG l-
0000D0	49 37 49 37 00 00 94 91 - 4C 35 C4 3C C6 33 03 00	I7I7 "L5Ä<Æ3
0000E0	E5 4D 47 5F 31 37 37 39 - 4A 50 47 20 00 2E 09 98	âMG_1779JPG . ~
0000F0	49 37 4C 37 00 00 D7 72 - 49 37 5E 3E 64 B2 4D 00	I7L7 xri7^>d²M
000100	E5 4D 47 5F 31 37 38 37 - 4A 50 47 20 00 97 0D 98	âMG_1787JPG - ~
000110	49 37 4C 37 00 00 84 75 - 49 37 38 65 D8 FC 36 00	I7L7 „uI78eØü6
000120	E5 4D 47 5F 31 37 37 38 - 4A 50 47 20 00 A2 10 98	âMG_1778JPG c~
000130	49 37 4A 37 00 00 D5 72 - 49 37 B7 80 3A 3A 41 00	I7J7 ÖxI7·c::A
000140	E5 4D 47 5F 32 33 38 30 - 4A 50 47 20 00 3F 4E A4	âMG_2380JPG ?N¤
000150	49 37 4A 37 00 00 14 9A - 30 35 44 16 DD AF 22 00	I7J7 š05DÝ" "
000160	E5 4D 47 5F 32 33 37 39 - 4A 50 47 20 00 21 51 A4	âMG_2379JPG !Q¤
000170	49 37 4C 37 00 00 12 9A - 30 35 9C 27 96 42 1F 00	I7L7 š05œ'-B
000180	E5 4D 47 5F 32 33 39 34 - 4A 50 47 20 00 05 87 A4	âMG_2394JPG ‡¤
000190	49 37 4C 37 00 00 15 7D - 38 35 55 A1 30 F6 35 00	I7L7 }85U;0ö5
0001A0	E5 4D 47 5F 31 38 33 31 - 4A 50 47 20 00 1C CA A4	âMG_1831JPG Ê¤
0001B0	49 37 4C 37 00 00 14 51 - CD 34 51 BC 7F D3 31 00	I7L7 Qí4Q*•Ó1
0001C0	E5 4D 47 5F 31 38 32 34 - 4A 50 47 20 00 09 CD A4	âMG_1824JPG Í¤
0001D0	49 37 4A 37 00 00 56 70 - CC 34 3B D5 CE 04 16 00	I7J7 Vpî4;Ôî
0001E0	E5 4D 47 5F 31 38 32 39 - 4A 50 47 20 00 46 CE A4	âMG_1829JPG Fî¤
0001F0	49 37 4C 37 00 00 D3 4E - CD 34 3E E0 94 5F 25 00	I7L7 ÓNí4>â"_%

Subdirectory

000000	2E 20 20 20 20 20 20 20 - 20 20 20 10 00 47 22 4D	.	G"M
000010	03 39 03 39 00 00 23 4D - 03 39 1D 53 00 00 00 00	99 #M9S	
000020	2E 2E 20 20 20 20 20 20 - 20 20 20 10 00 47 22 4D	..	G"M
000030	03 39 03 39 00 00 23 4D - 03 39 00 00 00 00 00 00	99 #M9	
000040	41 55 54 4F 45 58 45 43 - 42 41 54 20 00 4C 22 4D	AUTOEXECBAT	L"M
000050	03 39 03 39 00 00 94 73 - 27 1F 1E 53 E0 02 00 00	99 "s'sà	
000060	41 55 54 4F 45 58 45 43 - 44 43 53 20 00 51 22 4D	AUTOEXECDCS	Q"M
000070	03 39 03 39 00 00 45 7B - E3 1E 20 53 C1 02 00 00	99 E{ã SÅ	
000080	41 58 20 20 20 20 20 20 - 42 41 54 20 00 57 22 4D	AX BAT	W"M
000090	03 39 03 39 00 00 36 79 - 39 1C 22 53 57 00 00 00	99 6y9"SW	
0000A0	42 20 20 20 20 20 20 20 - 42 41 54 20 00 5C 22 4D	B BAT	\ "M
0000B0	03 39 03 39 00 00 25 61 - 6A 24 23 53 4C 00 00 00	99 %aj\$#SL	

```

0000C0 42 41 43 4B 53 45 57 20 - 42 41 54 20 00 62 22 4D BACKSEW BAT b"M
0000D0 03 39 03 39 00 00 E8 51 - 8F 38 24 53 8A 01 00 00 99 èQ•8$SŠ
0000E0 42 4B 20 20 20 20 20 - 42 41 54 20 00 67 22 4D BK BAT g"M
0000F0 03 39 03 39 00 00 73 6C - 82 25 25 53 37 02 00 00 99 sl,%%S7
000100 42 55 49 4C 44 41 4C 4C - 42 41 54 20 00 6B 22 4D BUILDALBAT k"M
000110 03 39 03 39 00 00 AE 7A - 8F 32 27 53 C8 0F 00 00 99 @z•2'SÈ
000120 43 43 35 20 20 20 20 - 42 41 54 20 00 72 22 4D CC5 BAT r"M
000130 03 39 03 39 00 00 48 8D - FA 38 2F 53 E8 00 00 00 99 H•ú8/Sè
000140 43 43 38 36 20 20 20 20 - 42 41 54 20 00 78 22 4D CC86 BAT x"M
000150 03 39 03 39 00 00 BA 94 - 89 28 30 53 1D 02 00 00 99 °"(OS
000160 43 43 53 43 35 20 20 20 - 42 41 54 20 00 7D 22 4D CCSC5 BAT }"M
000170 03 39 03 39 00 00 09 71 - 8A 1E 32 53 36 00 00 00 99 qŠ2S6
000180 43 43 5A 35 20 20 20 20 - 42 41 54 20 00 83 22 4D CCZ5 BAT f"M
000190 03 39 03 39 00 00 A0 5A - 5C 25 33 53 5A 00 00 00 99 Z\%3SZ
0001A0 43 46 44 4C 4C 20 20 20 - 42 41 54 20 00 86 22 4D CFDLL BAT †"M
0001B0 03 39 03 39 00 00 69 B5 - CA 24 34 53 0A 01 00 00 99 ipÊ$4S
0001C0 43 46 4F 52 4D 41 54 20 - 42 41 54 20 00 8E 22 4D CFORMAT BAT Ž"M
0001D0 03 39 03 39 00 00 7C 72 - 98 19 35 53 E1 01 00 00 99 |r~5Sá
0001E0 43 4F 4D 50 49 4C 45 20 - 42 41 54 20 00 92 22 4D COMPILE BAT ' "M
0001F0 03 39 03 39 00 00 23 07 - A6 1A 36 53 DD 00 00 00 99 #| 6SY

```

Each 0x20 byte entry used to be a file on early DOS versions, but now the structure has been enhanced (see later) to allow for names longer than 8.3. However, it is fully compatible and the first character in the entry can have two control values. If it is set to 0xE5 then the entry is a deleted file. If it is set to 0x00, then this is the end of the directory file. Any other value is the first character of the file name.

The two sectors above all have short 8.3 file names, all upper case.

The record structure is as follows

Bytes 0x00-0x0A File name as 8 characters then three characters extension. The '.' is automatically added

Byte 0x0B File attribute

-0-

NTFS directory entry, MFT

[Home](#)

NTFS creates directories from records in the \$MFT file. Each MFT record has a maximum length of 0x400 bytes (1024) and is always stored in two consecutive sectors.

The first sector always starts with FILE followed by a '0' or '*' depending on version of operating system

A very common location for the start of the \$MFT file is 0x60003F

When an MFT sector is viewed in CnW Recovery software, the sector is parsed, and a tool tip will display values for each type of field with the complete record

Sector 0x60003F

```

000000 46 49 4C 45 30 00 03 00 - F3 D4 36 9D 03 00 00 00 FILE0 606•
000010 01 00 01 00 38 00 01 00 - F8 01 00 00 00 04 00 00 8
000020 00 00 00 00 00 00 00 00 - 06 00 00 00 00 00 00 00
000030 93 04 00 00 00 00 00 00 - 10 00 00 00 60 00 00 00 "
000040 00 00 18 00 00 00 00 00 - 48 00 00 00 18 00 00 00 H
000050 40 39 E3 BE 35 B2 C6 01 - 40 39 E3 BE 35 B2 C6 01 @9ã¼5²E@9ã¼5²E
000060 40 39 E3 BE 35 B2 C6 01 - 40 39 E3 BE 35 B2 C6 01 @9ã¼5²E@9ã¼5²E
000070 06 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
000080 00 00 00 00 00 01 00 00 - 00 00 00 00 00 00 00 00
000090 00 00 00 00 00 00 00 00 - 30 00 00 00 68 00 00 00 0
0000A0 00 00 18 00 00 00 03 00 - 4A 00 00 00 18 00 01 00 J
0000B0 05 00 00 00 00 00 05 00 - 40 39 E3 BE 35 B2 C6 01 @9ã¼5²E
0000C0 40 39 E3 BE 35 B2 C6 01 - 40 39 E3 BE 35 B2 C6 01 @9ã¼5²E@9ã¼5²E
0000D0 40 39 E3 BE 35 B2 C6 01 - 00 80 38 2A 00 00 00 00 @9ã¼5²E €8*
0000E0 00 80 38 2A 00 00 00 00 - 06 00 00 00 00 00 00 00 €8*
0000F0 04 03 24 00 4D 00 46 00 - 54 00 00 00 00 00 00 00 $ M F T
000100 80 00 00 00 50 00 00 00 - 01 00 40 00 00 00 01 00 € P @
000110 00 00 00 00 00 00 00 00 - 87 A3 02 00 00 00 00 00 #£
000120 40 00 00 00 00 00 00 00 - 00 80 38 2A 00 00 00 00 @ €8*
000130 00 80 38 2A 00 00 00 00 - 00 80 38 2A 00 00 00 00 €8*
000140 33 24 D5 01 00 00 0C 43 - 64 CE 00 CB 11 F7 02 00 3$Õ Cđİ Ē+
000150 B0 00 00 00 A0 00 00 00 - 01 00 40 00 00 00 05 00 ° @
000160 00 00 00 00 00 00 00 00 - 15 00 00 00 00 00 00 00 @
000170 40 00 00 00 00 00 00 00 - 00 60 01 00 00 00 00 00 @
000180 C8 51 01 00 00 00 00 00 - C8 51 01 00 00 00 00 00 ÈQ ÈQ
000190 31 01 FF FF 0B 31 01 A4 - 5E 70 31 05 40 82 E5 41 lÿÿ 1¤^p1@,âA
0001A0 01 D8 F4 F0 00 31 01 D5 - 44 03 41 01 A9 C4 7E FF Øóð 1ÖDA@Ä~ÿ
0001B0 31 01 69 79 7A 41 01 60 - 63 85 00 31 01 25 4E 01 liyzA`c... 1%N
0001C0 31 01 A6 62 01 31 01 20 - 7F 01 41 01 EA 54 77 FF 1;b1 •A êTwÿ
0001D0 31 01 95 32 04 31 01 A6 - EE 7A 31 01 1C 00 03 31 1•21;îz1 1
0001E0 01 84 47 10 31 01 DD E2 - 0F 31 01 31 AF 0A 00 00 „G1Ýâ11
0001F0 FF FF FF FF 00 00 00 00 - FF FF FF FF 00 00 93 04 Ÿÿÿÿ Ÿÿÿÿ "

```

Next sector, 0x600040

```

0001A0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001B0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001C0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001D0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001E0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
0001F0 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 93 04 "

```

To help recognise an MFT sector it should be noted that the last two bytes of each sector will always be the same. These bytes are set with a 'random' value that is then modified later. It ensures that both sectors have been read

fully. In the example above, one can note that the final two bytes are 93 04 and these values are also set in bytes 0x30-0x31 to show the value that should be read

With CnW software, when the sector is viewed with View Sector, as the cursor is moved over each field in the MFT record, it will be decoded and displayed as a tool tip. Most useful values can be the date fields and size fields that are not always obvious, or easy to decode.

-0-

Disk clusters

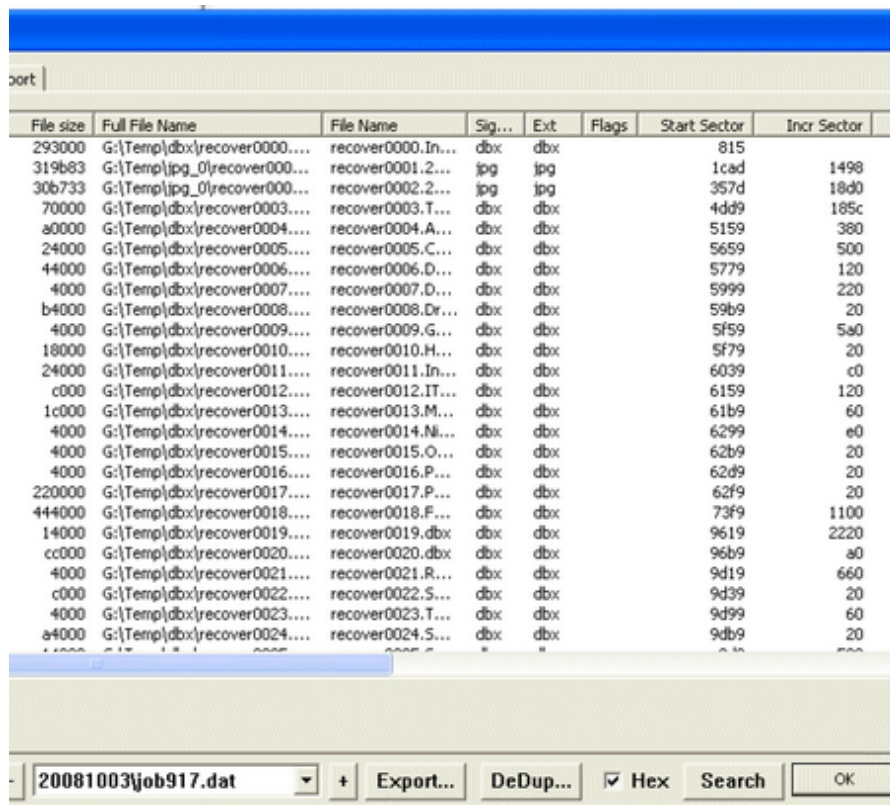
[Home](#)

Disks consist of sectors, normally 512 (0x200) bytes in length. With a modern, 500GB disk, this means there are about 1,000,000,000 separate sectors that the operating system has to manage. As these numbers can get rather large, operating systems work in groups of sectors, and call it a cluster. A cluster is then the smallest amount of disk that can be allocated, and are always contiguous runs of sectors.

The size of a cluster is always a compromise. A large cluster means that there are fewer clusters for the operating system to manage, but there is always the problem that a small file will require a complete cluster, and so can represent a large amount of wasted space. A small cluster reduces the amount of wasted space, but will require many more to be tracked by the operating system.

When recovering a disk it is often useful to know the size and location of clusters. If the disk has a valid operating system, then this will be determined from information within the [BPB](#). If the operating system information is lost - or maybe not valid due to a reformat - then it will be necessary to determine the cluster size, and location. There are built in tools for FAT and NTFS disks to try and determine values, but the other way is to examine the log after an Image Raw Scan.

Once a Image Raw scan of a disk is done, the log should be opened, and the data viewed in hex mode



File size	Full File Name	File Name	Sig...	Ext	Flags	Start Sector	Incr Sector
293000	G:\Temp\dbx\recover0000....	recover0000.In...	dbx	dbx		815	
319b83	G:\Temp\jpg_0\recover000....	recover0001.2...	jpg	jpg		1cad	1498
30b733	G:\Temp\jpg_0\recover000....	recover0002.2...	jpg	jpg		357d	18d0
70000	G:\Temp\dbx\recover0003....	recover0003.T...	dbx	dbx		4dd9	185c
a0000	G:\Temp\dbx\recover0004....	recover0004.A...	dbx	dbx		5159	380
24000	G:\Temp\dbx\recover0005....	recover0005.C...	dbx	dbx		5659	500
44000	G:\Temp\dbx\recover0006....	recover0006.D...	dbx	dbx		5779	120
4000	G:\Temp\dbx\recover0007....	recover0007.D...	dbx	dbx		5999	220
b4000	G:\Temp\dbx\recover0008....	recover0008.Dr...	dbx	dbx		59b9	20
4000	G:\Temp\dbx\recover0009....	recover0009.G...	dbx	dbx		5f59	5a0
18000	G:\Temp\dbx\recover0010....	recover0010.H...	dbx	dbx		5f79	20
24000	G:\Temp\dbx\recover0011....	recover0011.In...	dbx	dbx		6039	c0
c000	G:\Temp\dbx\recover0012....	recover0012.IT...	dbx	dbx		6159	120
1c000	G:\Temp\dbx\recover0013....	recover0013.M...	dbx	dbx		61b9	60
4000	G:\Temp\dbx\recover0014....	recover0014.Ni...	dbx	dbx		6299	e0
4000	G:\Temp\dbx\recover0015....	recover0015.O...	dbx	dbx		62b9	20
4000	G:\Temp\dbx\recover0016....	recover0016.P...	dbx	dbx		62d9	20
220000	G:\Temp\dbx\recover0017....	recover0017.P...	dbx	dbx		62f9	20
444000	G:\Temp\dbx\recover0018....	recover0018.F...	dbx	dbx		73f9	1100
14000	G:\Temp\dbx\recover0019....	recover0019.dbx	dbx	dbx		9619	2220
cc000	G:\Temp\dbx\recover0020....	recover0020.dbx	dbx	dbx		96b9	a0
4000	G:\Temp\dbx\recover0021....	recover0021.R...	dbx	dbx		9d19	660
c000	G:\Temp\dbx\recover0022....	recover0022.S...	dbx	dbx		9d39	20
4000	G:\Temp\dbx\recover0023....	recover0023.T...	dbx	dbx		9d99	60
a4000	G:\Temp\dbx\recover0024....	recover0024.S...	dbx	dbx		9db9	20

The important parameters are the start sector value, and Incr(ement) sector, and the reason for viewing in hex is that all clusters sizes are multiples of 2, ie 1,2,4,8,16 etc.

In the example above it can be seen that the majority of increments are multiples of 0x20 (ie 32). With an Image Raw, there may be false positive starts detected, and so it can be seen that the top few values do not fit the pattern. For the majority of the files, it looks safe to say that the cluster size is 0x20

Looking at the Start Sector it is very clear that the majority of files start with a sector value ending in 9. The first cluster of a disk can be located on any sector location, and so for a cluster size of 0x20, the start of a cluster could in theory be any value between 0x00 and 0x1F. In the example above, files always start at vlaue such as 0xb9, 0x19, 0x39 so the start value of the cluster would be sector 0x19.

Apple Volume Header

[Home](#)

This sector is normally 2 sectors after the start of the partition - for HFS Plus disks, it always starts with 'H+'. A typical sector number for this sector is 0x6402a

00000000	48 2B 00 04 00 00 60 00 - 48 46 53 4A 00 00 AF 01	H+ ` HFSJ -
00000010	C1 E6 80 12 CC 0E 49 C7 - 00 00 00 00 C1 E6 F0 92	ÁæĲİ İÇ Áæø'
00000020	00 0A F3 E1 00 02 6E AD - 00 00 10 00 00 DE 47 86	óá n ßG†
00000030	00 1C DB D5 00 75 68 12 - 00 01 00 00 00 01 00 00	ÛÖ uh
00000040	00 BE 3E C1 29 C9 26 0B - 00 00 00 00 02 00 00 8B	¼>Á) É& <
00000050	00 00 0C 8A 00 9B 1A 14 - 00 00 00 00 00 00 00 00	Š >
00000060	00 00 00 00 00 00 0C 8A - 19 BD 42 F8 8A 51 5A 3A	Š½BøŠQZ:
00000070	00 00 00 00 00 00 80 00 00 - 00 1B D0 00 00 00 08 00	€ Đ
00000080	00 00 00 01 00 00 08 00 - 00 00 00 00 00 00 00 00	
00000090	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
000000A0	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
000000B0	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
000000C0	00 00 00 00 00 50 00 00 - 00 60 00 00 00 00 05 00	P `
000000D0	00 00 08 01 00 00 05 00 - 00 00 00 00 00 00 00 00	
000000E0	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
000000F0	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000100	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000110	00 00 00 00 17 60 00 00 - 01 30 00 00 00 01 76 00	` 0 v
00000120	00 00 0D 01 00 00 9A 00 - 00 01 3D D2 00 00 0E 00	š =Ö
00000130	00 41 06 D3 00 00 A8 00 - 00 6B D0 45 00 00 13 00	ÁÓ " kÐE
00000140	00 93 92 4F 00 00 13 00 - 00 00 00 00 00 00 00 00	"'O
00000150	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000160	00 00 00 00 00 7F 00 00 - 01 30 00 00 00 00 07 F0	• 0 ð
00000170	00 4F E8 87 00 00 07 F0 - 00 00 00 00 00 00 00 00	Oè† ð
00000180	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	

The sector is extremely important for recovery as it has pointers to both the catalog, and extents table. It also has the basic information regarding cluster sizes. The sector is normally duplicated near the end of the partition

Important fields in the volume header

Bytes 0x00-0x01	Signature, H+
Bytes 0x02-0x03	Version, 0x4
Bytes 0x10-0x13	Create date
Bytes 0x14-0x17	Modify date
Bytes 0x18-0x1B	Backup date
Bytes 0x1c-0x1F	Checked date
Bytes 0x20-0x23	File count
Bytes 0x24-0x27	Folder count
Bytes 0x28-0x2B	Block size, typical 0x1000
Bytes 0x2c-0x2F	Total blocks
Bytes 0x30-0x33	Free blocks
Bytes 0x38-0x3B	Resource clump size
Bytes 0x3C-0x3F	Data clump size
Bytes 0xc0-0x10f	Extents file locations

Bytes 0x110-0x15f Catalog file locations

-0-

VMFS sectors

[Home](#)

VMFS is a Virtual file system from VMWARE. CnW Development in underway to produce a flat file from the virtual format.

Several sectors have fixed patterns and are shown below

Partition entry block

00000000	FA B8 00 10 8E D0 BC 00 - B0 B8 00 00 8E D8 8E C0	ú, ŽĐ¼ °, ŽØŽÀ
00000010	FB BE 00 7C BF 00 06 B9 - 00 02 F3 A4 EA 21 06 00	û¼ ¿ ¹ ó¼ê!
00000020	00 BE BE 07 38 04 75 0B - 83 C6 10 81 FE FE 07 75	¼¼ø u fÆþbu
00000030	F3 EB 16 B4 02 B0 01 BB - 00 7C B2 80 8A 74 01 8B	óë´ ° » ²eŠt<
00000040	4C 02 CD 13 EA 00 7C 00 - 00 EB FE 00 00 00 00 00	LÍê ëþ
000001A0	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
000001B0	00 00 00 00 00 00 00 00 - 4B 06 0A 00 00 00 00 02	K
000001C0	03 00 FB FE FF FF 80 00 - 00 00 FC 9E B6 2D 00 00	ûþýýe üžŒ-
000001D0	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
000001E0	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
000001F0	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 55 AA	Uª

Offset 0x1c6 is the start location of the virtual disk

Offset 0x1ca is the end of the virtual disk - both values in 0x200 byte sectors

Info block

00000000	5E F1 AB 2F 04 00 00 00 - 2E C9 3E 5F 3C C0 6E 00	^ñ</ .É> <Än
00000010	FC 5E 5F 68 B5 99 C9 53 - AD 02 00 00 00 64 61 74	ü^_hµ™ÉS dat
00000020	61 73 74 6F 72 65 32 00 - 00 00 00 00 00 00 00 00	astore2
00000030	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000040	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000050	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000060	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000070	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000080	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000090	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 02 00	
000000A0	00 00 00 20 00 00 00 00 - 00 C9 3E 5F 3C 01 00 00	É> <
000000B0	00 9A 3E 5F 3C EC 03 E2 - BD 55 08 68 B5 99 C9 53	š>_<iâ¼Uhµ™ÉS
000000C0	AD 01 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	

Magic numnber is 0x5ef1ab25 (Big endian)

Offset 0xa1 is the block size, ie 0x200000

VMFS Header, LVM

00000000	0D D0 01 C0 03 00 00 00 - 11 00 00 00 02 16 01 00	ĐÀ
00000010	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
00000020	00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 60 05	,
00000030	08 B1 00 10 52 39 53 59 - 41 4B 4E 55 02 00 4C 4F	± R9SYAKNU LO
00000040	47 49 43 41 00 00 00 00 - 00 00 00 00 00 00 00 00	GICA
00000050	00 00 00 00 00 00 00 00 - 00 00 00 02 00 00 00 F8	ø
00000060	3D 6D 5B 00 00 00 01 00 - 00 00 B6 05 00 00 B5 05	=m[Œ µ
00000070	00 00 03 00 00 00 00 00 - 00 00 00 00 10 01 00 00	
00000080	00 00 9E 3E 5F 3C F5 3E - ED 82 8D 2E 68 B5 99 C9	ž> <ö>i,•.hµ™É
00000090	53 AD EF 86 3C 51 34 99 - 03 00 AE E6 CF 53 34 99	Si†<Q4™ @æİS4™
000000A0	03 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00	
000000B0	00 00 00 00 00 00 00 00 - 00 00 A9 FE 00 01 00 00	©p

```

00000200  00 00 00 60 5B 00 00 00 - B7 05 00 00 00 00 00 00  `[ .
00000210  01 00 00 00 33 63 35 66 - 33 65 39 61 2D 62 64 65  3c5f3e9a-bde
00000220  32 30 33 65 63 2D 30 38 - 35 35 2D 36 38 62 35 39  203ec-0855-68b59
00000230  39 63 39 35 33 61 64 00 - 00 00 00 00 00 00 00 00  9c953ad
00000240  00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
00000250  00 00 00 00 9A 3E 5F 3C - EC 03 E2 BD 55 08 68 B5  š> <iâ½Uhp
00000260  99 C9 53 AD 01 00 00 00 - 4A 31 3D 51 34 99 03 00  ™ÉS J1=Q4™
00000270  00 00 00 00 B6 05 00 00 - 00 00 00 00 00 00 00 00  ℥
00000280  B5 05 00 00 00 00 00 00 - 3B 65 CF 53 34 99 03 00  μ ;eİS4™
00000290  01 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00

```

Magic number is 0x0dd001c0 (Big endian)

-0-

General Tools

[Home](#)

CnW Recovery software has several functions that can be used once files have been recovered from corrupted media. Typically they assist in Video disk recovery, and also processing files for writing to DVDs

Tools are accessed from the Tools entry in the top drop down menu

[Split directories for DVD burning](#)
[Rebuild video disk from MPEG files](#)
[Merge disk images](#)
[User Passwords](#)

-0-

Split directories

[Home](#)

Split directories will allow a directory tree to be split into sections suitable for burning to DVDs. This function can be used when it is necessary to transfer many files to DVDs and will try and fill DVDs to approx 90-95% capacity.

It is possible to recover files into DVD sized directories when doing recovery, but this can often mean that directories are scattered over several DVDs. This occurs when certain recovery modes are use, such as NTFS Recover from file entries. In order to produce DVDs with a more logical directory structure, files should be recovered without the DVD option, and then this split tool can be run.

To run the function, it is just necessary to enter the name of the main directory storing the files to be placed into DVD directories. The output files will be moved into new directories on the same drive, in a directory with a 0 added to the original directory name. Thus if the initial directory is "SplitDVD" the output directory will be "SplitDVD0". As the files are moved, the original file will disappear, and so files can be split on a full drive, very little extra space is required.

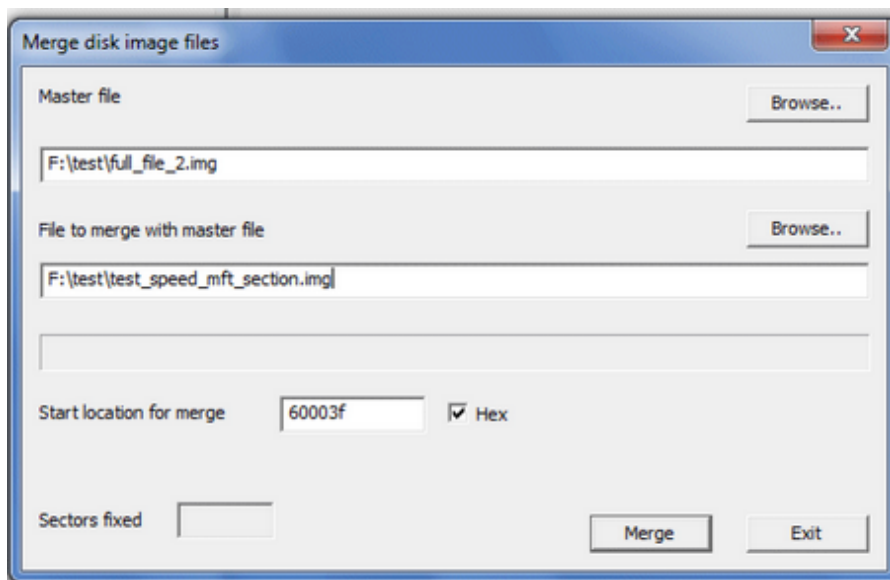
-0-

Merge disk images

[Home](#)

Occasionally one has duplicate copies of a logically identical media. However, at time each copy has different failures.

The merge disk images allows two disk images to be merged. The routine will analyse the master file, and if a blank sector is found, it will then test the second file. If the second file has a non blank sector, this will be merged into the master file.



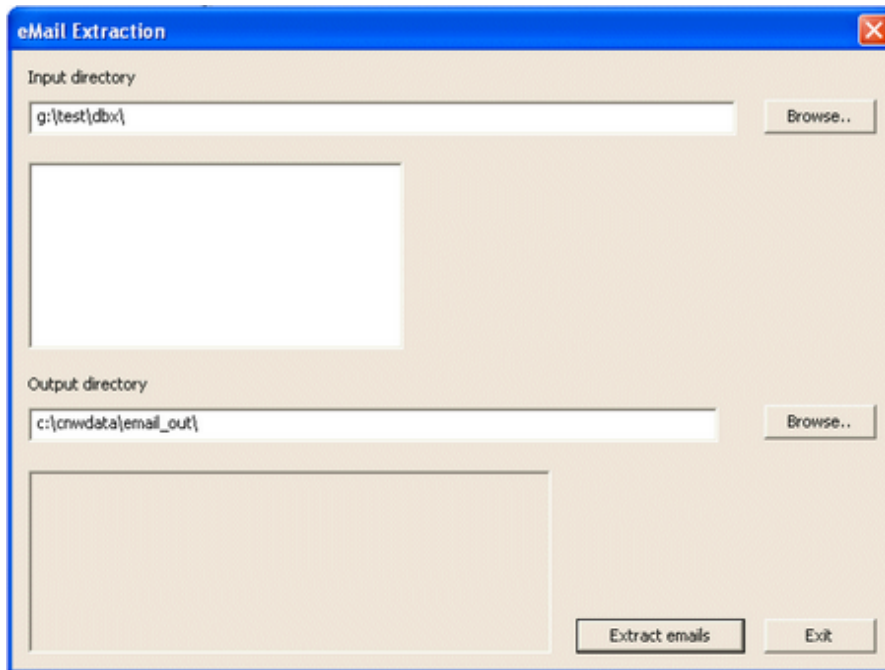
There are occasions where one wants to merge a partial file with a full file. If for instance, as in the example shown above, there is an image of just the start of MFTs, this will be merged into the main file starting at sector 0x60003f. It is not possible to merge a file that starts after the end of the master file.

-0-

eMail Extraction

[Home](#)

eMail extraction is a tool to help convert certain type of e-mail file into text copies of e-mails. It will operate with Outlook Express 6, dbx files and also Macintosh Database files. It does assume that the file is largely intact.



There are two parameters that have to be set, the location of the input files, and where the output files are to be saved. The program will automatically detect file format and extract the e-mails. When possible, the output e-mail will have a name based on content and date and time .

-0-

User passwords

[Home](#)

Many disks are protected by passwords controlling access to the disk and often user files. CnW Recovery software reads files directly and by passes all such passwords. It is therefore not necessary to have access to a password in order to read files from a directory that the operating system will not allow access to.

The same applies to system files, which can all be read and saved on a new hard drive.

User login passwords

Sometimes (typically a laptop) can not be used because the user has defined a password that has since been lost, or deliberately changed. As described above, CnW Recovery software will read the files, but it may be necessary to find the password in order to handle encrypted files and to restore the PC to a full working state. CnW does not do any decoding of passwords but it can read two of the critical files that store the information. These are the SAM file and SYSTEM file, typically stored in windows\system32\config It is not normally possible to read such files without special tools, or CnW. CnW does require the hard drive operating as a slave drive to access such files.

There are then several tools that may be found on the web to assist with accessing passwords, and two such programs are 'John the Ripper' and SAMInside, which both make use of the SAM and SYSTEM files. A web search will also highlight many more.

-0-

AVCHD reconstruction

[Home](#)

AVCHD is a high definition video standard now being used by quality cameras and video recorders. When a raw recovery of data is performed, several types of files are found. The most important one is the MTS file which stores the video data. Other files are .CPI, MPL etc. For a video editor to process these, they have to have relevant names, and be stored in valid directories. This tool will assist in the process.

-0-

Extract and join

[Home](#)

This function allows sections of a file to be extracted and a new file created from these fragments. It can be a useful tool if doing manual data carving, and wanting to reconstruct a file.

A fragment can be any length of sectors. To aid with file creation, the location that the fragment is stored in can be defined (as sector offset).

The utility is used by entering the start sector, and then either the length or end sector. If the offset is set as zero, then the section will be appended to the previous section.

-0-

Fake memory test

[Home](#)

Some memory chips, (eg sD, CE or USB memory sticks) are marked and sold as one capacity, but are infact much smaller in size.

The chips will format correctly, and give every indication that they are size marked on the case. In fact the firmware within the chip is normally 'fiddled with' to make sure they look genuine

When used, a few photos or video will be stored correctly, but maybe 90% on a full memory chip will be lost. This tool is designed to test memory chips to ensure that they are valid. There are two tests.

Test 1

The first test is to examine for data at the end of the memory chip and see if it is repeated earlier in the memory. If this pattern is found then it unfortunately indicates that it is too late, and data has been lost

Test 2

The second test is only required if there is no data written at the end of the physical chip, such as when the memory chip is new. In this test, a sector with a unique pattern is written near the end of the physical memory. The memory chip is then examined to see if this new data occurs else where on the memory chip. If the pattern is found again, then this is a fake memory chip, if it is not found, then the chip is valid.

-0-

Reconstruction tips

[Home](#)

Once files have been recovered, they often need to be restored back to a new operational disk. For most, this is a straight forward copy into the relevant directories, but others require a bit more effort.

This section gives some useful tips

[eMail Restoration - Outlook Express 6](#)

-0-

eMail restoration

[Home](#)

Outlook Express 6

The e-mail files for Outlook Express 6 have the file extension of .DBX. Files may be found by file recovery modes, or by Raw recovery. Once found, they cannot just be copied to a directory, they have to be imported with Outlook.

To import the files, they need to be placed in a subdirectory which must not be on a DVD or CD-ROM. They also need to include with the files, the file Folders.dbx. Without this file, the error message 'No messages or files can be found in this folder or another application is running that has the required files open'.

The import procedure is a series of Outlook Express menu operations as follows,

File / Import / messages... / Outlook Express 6 / Next/ Browse..

at this point browse to the location of the subdirectory where .DBX to be imported are files are.

After Next, the details of the main boxes to be imported are shown, and may be selected.

-0-

Logs

[Home](#)

CnW Logs provide details of all jobs done. They are an essential part of any forensic investigation, as well as very useful as a summary for any disk recovery.

- [Log overview](#)
- [File details](#)
- [Search for sector](#)
- [File fragments](#)
- [Job details](#)
- [Forensic Report](#)
- [Trace?.txt file](#)

-0-

Log overview

[Home](#)

The log stores details of all files that have been read, along with media errors. There is also an option to export the log into a CSV file. The default file name will be the job number of conversion. The details stored in the log do depend on the options purchased with the CnW software package. The standard package will have file names, dates and sizes. The full logging option, for forensic investigations, will contain details on file locations, as well as a MD5 sum check for each file read.

The log has four sections

[File detail](#)

[Details about the media](#) and the job

[Forensic Log](#) (Forensic option only)

[Keyword search Log](#) (Commerical and Forensic only)

An important feature of the log is that it is possible to double click on a file name and view an image of the file. Current versions will just display images, under development is a hex and text display of data. This works in demo mode, so even though files can not be restored to a hard disk, the image that would be recovered, can be displayed.

The file view screen will either display a picture, or will display a hex dump of the first 1MB of the file. Any picture image displayed is stretched to the size of the window – this may distort the image, and so should only be taken as an indication, rather than a valid image. This will give a good indication to see if the file is correct

-0-

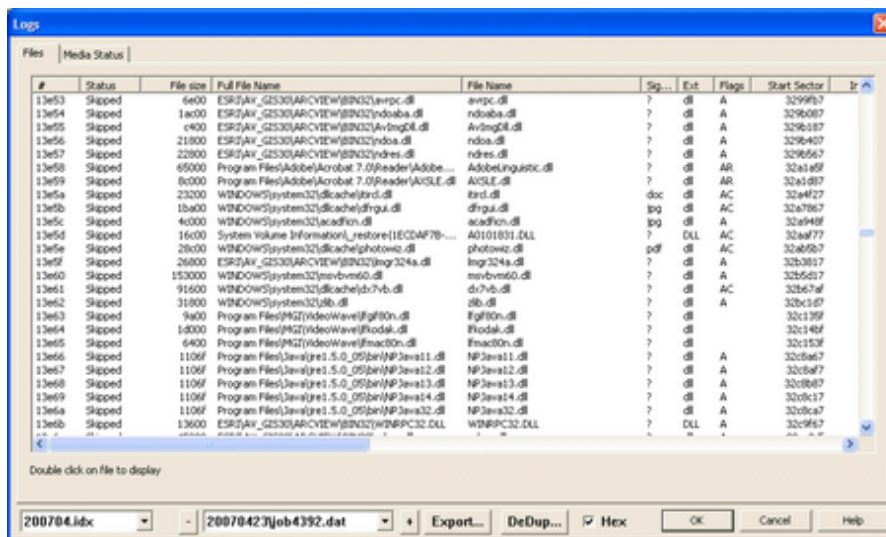
File details displayed in the log

[Home](#)

The following data is displayed in the log. It is slightly media dependant and some items are only available with the Full Log option – designed for forensic investigators and other power users.

From the log screen, previous logs may be displayed, by when ever the log screen is started, the most recent log is displayed.

Any column may be sorted by clicking the column header - there is currently (April 2007) a (large) limit on the maximum number of entries that can be sorted, but that will be removed in later versions.



All sector numbers are absolute numbers on the disk drive, and not relative to the current partition

Log selection

Logs are stored with an incrementing job number. Also, to help with house keeping, they are grouped on a monthly basis. Both these are controlled by the combo boxes at the bottom of the display. The left hand box selects the month, so viewed above, it is February 2006. The second combo box selects the actual log. It is stored in a subdirectory, with a name generated from the date. In the example above, it is 2 February 2006, the format is YYYYMMDD. The job number will 'never' roll over.

To job between consecutive logs, the + and - buttons may be used.

consecutive

Status

This describes the contents of the log record and can have many possible values, listed below.

- Del The file was detected in the directory as a deleted file.
- Directory - this is the name and location of a directory found
- Error – an error was detected
- IC - The file is incomplete, ie less recovered than stated in directory. For a FAT disk this is often a corrupted FAT
- MFT The file was recovered correctly, it was based on an MFT in NTFS
- OK The file has been copied correctly
- Over The file that has been read contains sectors that have been read in other files. ie the original file has probably been partially or completely overwritten.
- Rec'vd The file has been recovered by using the read unallocated space, or the disk image scan. These files may or may not be valid and should be tested before assuming they are complete or correct.
- Scan The file entry has been found by scanning the directory, no copy has been made
- Skipped The file has been skipped, either because it is a deleted file, or because of the file filter options

It should be noted that the status will often change after a restore has been made after a scan. A Scan will normally have the status of Scan, rather than OK etc

File size

This is the filesize as read by the directory entry. ie, if the file fails to read completely, the log shows the expected filesize

Filename

This is the filename as read from the disk directory

Signature

The signature is determined by analysing the start of the data. For a scan, there is no signature test, and so remains as unknown, or '?'. Many file types have the same start of a file, so for instance a DLL, EXE both start with the same codes, and so the signature will always show as .exe. Experience is often required to determine if any differences between signature, and Extension(see next item) are significant. Files such as jpeg, or jpg have a unique start, so if the signature is detected as jpg, but the extension is marked as .doc, or .dat it could be a case deliberate renaming of a files, possibly to hide them.

When a signature is not recognised the first two hex values are displayed, eg 0x59 3F This helps see the start of the file, and if there is a possible pattern,

it will be clear.

Flags

The flags are values stored in the disk directory. When a flag, or attribute is detected, a letter is output in the field. These values are as follows

- A The archive flag is set
- C The file is compressed by the operating system, such as NTFS
- D The file was deleted
- E The file has been encrypted
- H The file is hidden
- R The file is read only - ie write protected
- S This file is a system file

2,3 etc This number is the number of streams that have been found. A single stream is not shown, and so counting starts at 2

Start sector

The start sector is the first sector of the file

Dir Sect

The directory sector is where the directory information is stored. For an NTFS this will be the address of the MFT block. By clicking on this column the the sector will be displayed

Parent Dir Sect

The parent directory sector is the location of the parent directory of the file

Dir offset

For many file systems, multiple file entries are stored within a larger directory file. The Directory offset is the location within the file for the particular file entry. For NTFS disks, the offset is the number of the MFT within the \$MFT file

Sect Err

If a missing or failed sector is encountered within a copy, then this flag will be set to yes. On some occasions the file will still open, but it will be corrupted. When sorted, it will be sorted along with the start sector number. This can often give an indication of areas of a disk that have failed, or in the case of a disk image, have not been imaged.

Frag

Many large files are fragmented on a disk, ie not written as a single continuous stream. The fragment count indicates how many fragments there are within the file. By clicking on this column, details of each [fragment](#) will be displayed.

Verify

The verify column will indicate Yes or No indicating if the file has passed several validation checks. These results must be treated with a bit of caution

as there can be false positives and negatives, sometimes due to changes in the original file structure due to program updates. If the verify column is clicked on, there is the tool for [Manual Data Carving](#) (as a forensic option only).

Filtr

The filter flag is Y or N. If Y the file has been copied, and if N, the file filter testing has meant that the file has not been copied

Create, Modify, and Access dates

These are the dates that file was created, modified, or accessed. It should be noted that the definition of modified, is when the contents have been changed. Creation is when the file was placed on the disk. It is therefore possible, if a file is moved from one location to another location on a different disk, that the creation date will be newer than the modified date.

MD5

This is an industry standard hash value of the file contents. If any single bit of the file is changed, the hash value will be different. No two files will ever have the same MD5 value. The file name and date are not part of the MD5 value. Forensic investigation often makes significant use of hash values.

It has two main uses,

- It can be used as a quick way to test that two files are identical, even if in different locations

- It can verify that the file has not been changed

The MD5 value is also used within the [file filter](#) to test files against a known database

Log time

The log time is the time that the log entry was created

Export

The export function will output all the log in a .CSV format. The location is defined by the values in the [Directories](#) configuration.

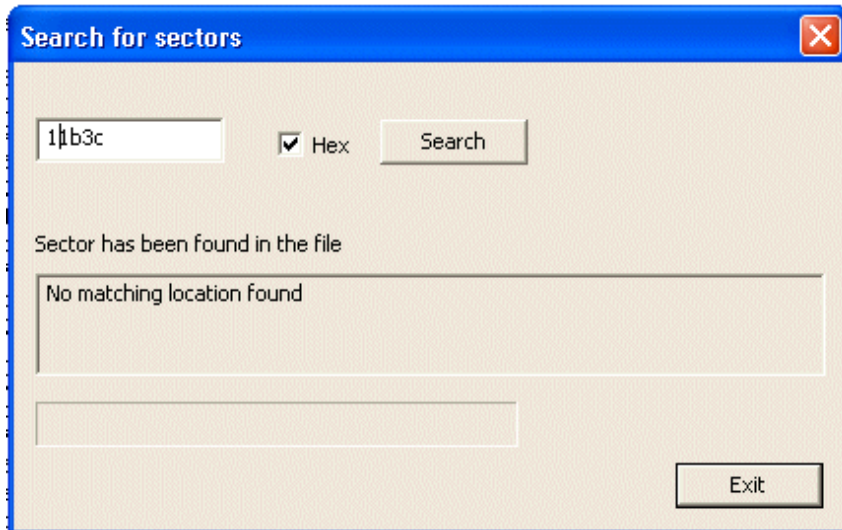
Search

The [search](#) function will show which file a sector is used in

Search for sector

[Home](#)

A very useful forensic tool is to determine which file a sector belongs to. In the case of overwritten, or deleted files, a sector may have more than one apparent owner



The value of the absolute location of the sector is entered into the box (in either hex or decimal according to the flag) and when Search is pressed the log is examined to determine which file(s) the sector is part of.

Obviously, a sector should only be used in a single file, but if deleted files have been restored within the log, these will be tested as well. If a deleted file has been overwritten, it should be possible to see which file overwrote it.

The routine will search up to 80 fragments on a file.

As a double check, when a file has been isolated, it is possible to view the fragments of the file by clicking on Frags column within the log.

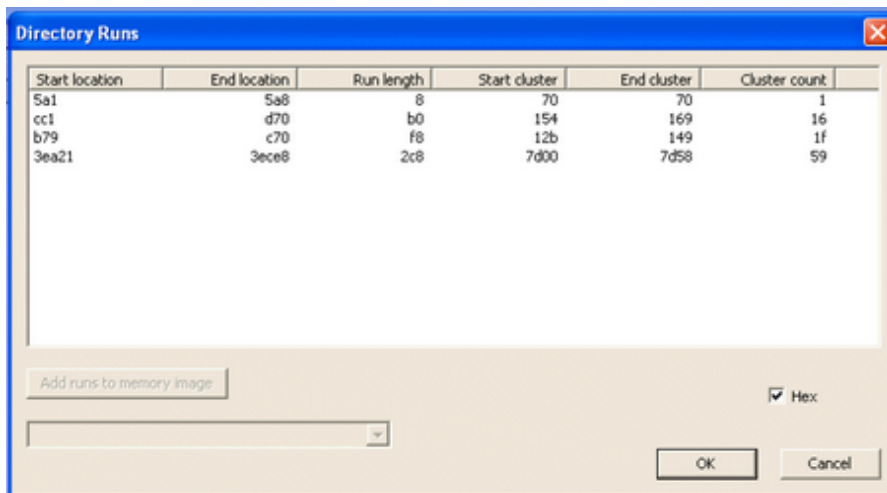
It must be noted that the log is only valid after a file recovery has been run. However, to save time, and space if recovery is not actually required, the solution is to use the 'Select Files' rather than Recover All. The disk will be scanned and this stage will be complete when the 'Select All' and 'Copy' buttons become enabled. The sector number may then be entered and matching file(s) displayed. If not found, then the sector is in the unallocated area.

File fragments

[Home](#)

Most files are stored on a disk in contiguous sectors. If a file is very large, or the disk is very full, then a file may be stored in many fragments. This is also very true when a file has been created by added small sections to file, as often happens with logs.

The CnW program will indicate in the log the number of fragments a file has. By double clicking on the fragment number in the log, details of the fragments will be displayed showing the start location, and length of the fragment.



The screenshot shows a window titled 'Directory Runs' with a table of file fragments. The table has six columns: Start location, End location, Run length, Start cluster, End cluster, and Cluster count. There are four rows of data. Below the table is a button 'Add runs to memory image', a checkbox 'Hex' which is checked, and a dropdown menu. At the bottom right are 'OK' and 'Cancel' buttons.

Start location	End location	Run length	Start cluster	End cluster	Cluster count
5a1	5a8	8	70	70	1
cc1	d70	b0	154	169	16
b79	c70	f8	12b	149	1f
3ea21	3ece8	2c8	7d00	7d58	59

The report shows both start and end location of each data run as well as the length. To assist in analysing disks, the information is shown in both sectors and clusters.

If the sector number is Start location or End location is double clicked, the sector will be displayed.

Job details

[Home](#)

The job details are obtained by clicking on the Media Status tag. This screen will give details of the media, and the job that are fixed for the media, rather than individual file details.

The screen is universal for all types of media, so not every field will be filled in for every media type.

The screenshot shows a software window titled 'Logs' with a 'Media Status' tab selected. The window contains several input fields and checkboxes for media details:

- Volume ID:** NEW
- Format:** Joliet
- Media type:** CDR/ROM
- Drive:** Memorex 48MBX 244816A3 KWHB 48B
- Library slot:** (empty)
- Capacity:** 737,275,904
- Erasable:** ☐
- Tracks/partitions:** 12.6
- Serial Number:** 0
- Mode:** Yellow Mode 1
- Finalized:** ☐
- Error count:** 0
- Retry count:** 0
- Copied:** 104,080
- Output path:** f:\test2\
- Layers:** 1
- Sessions:** 3

Below these fields is a section titled 'Track / session details' containing:

- File count:** 44
- Total file byte count:** 104,080
- Creation date:** 2006-04-25 19:40
- Session number:** (empty)
- Track start sector:** 0
- Track end sector:** 12000
- UDF VAT:** 0

At the bottom of the window, there are two dropdown menus showing '200605.idx' and '20060508\job3204.dat', followed by an 'Expert...' button and 'OK', 'Cancel', and 'Help' buttons.

Volume ID

This is the name read from the media. It is normally user entered when the media is formatted, or for a CD, when it is burnt.

Format

This is the logical format, or operating system. One will expect values such as NTFS, FAT32, UDF.

Media Type

This is the physical media. Values such as CDR/ROM, Floppy, and Hard drive will be seen. It is very difficult to distinguish between a hard drive and a flash memory chip, so they are described as Removable drive.

Drive

This provides details of the physical drive being used to read the media. For forensic inquiries this can be important. When reading a hard drive, the details stored here are those of the drive.

Library Slot

The library slot value is only used when media (eg CDs) are being read

from a random access library.

Capacity

The capacity is the raw capacity of the media, displayed in bytes. This does not define the amount of information on the media, and when compression is used, the data stored can exceed the stated capacity.

Erasable

The erasable flag is only relevant for CDs and DVDs. If set, then the media is RW (Read/Write) rather than write once.

Tracks partitions

Many types of media can be divided into logical smaller sections. For a hard drive, they can be partitioned so that they behave like completely separate drives. For CDs, the disk can be written as separate tracks. For audio CDs, this is used for separate songs etc. With data CDs, tracks are normally an indication that the CD has been written more than once, as a multi-session CD. However, it should be noted that a single session may contain multiple tracks, though typically, tracks and sessions map one to one to each other.

Serial Number

Many pieces of media have a unique serial number, added at creation time, or first initialisation. This is not normally user changeable.

Mode

CDs can be written in several different physical ways. Although these are largely invisible to any user, they can affect compatibility. The options that are recognised are

- CD Audio
- Yellow Mode 1
- Yellow Mode 2
- XA Mode 2, form 1
- XA Mode 2 form 2

Finalised

CDs can be appended to or have contents fixed. When fixed, the session is finalised, or closed and no additional sessions may be added.

Error Count

This is the number of errors detected when reading the media. Many disks can be read very successfully with several errors, but problems can occur when the error count goes into hundreds

Retry Count

This is the number of times the drive has to retry to read a sector. A large number of retries, and a small error count indicates a drive that is failing, media that has problems, or a drive media combination that is

not very compatible, or misaligned.
Copied

Output Path

The output path is where the files have been copied to. At times, depending on recovery mode, the output path may be appended with fixed directories such as !recover or !deleted.

Layers

Layers is for DVDs only. Some DVDs have 2 layers - and hence a capacity of about 9GB a side.

Sessions

For CDs there can be multiple sessions of writing. Typically each session on a data CD will be a new track, but there is no reason why sessions should not contain more than one track.

-0-

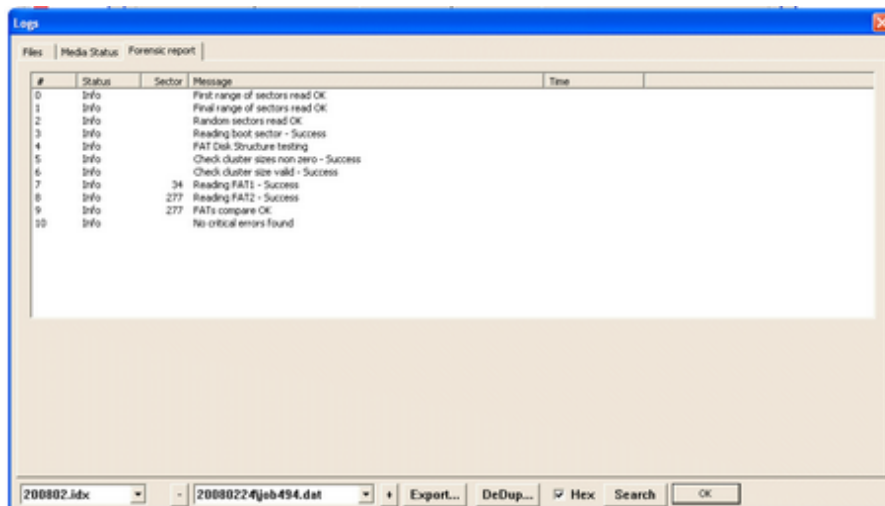
Forensic Report

[Home](#)

The forensic report is part of the Forensic Options, controlled by the program licence.

The function of the report is to be the basis of any log or report listing actions and tests on the drive. It will report any disk errors, and other errors the standard logical recovery have not been possible. The best example would be when it has not been possible to resolve a full directory path, and a dummy directory entry has been created.

The report is part of the main log screen, and selected by the tab at the top of the box.



Log entries, from program Wizard

The wizard performs tests on the physical media, and the following status messages, or errors will be detected. The results from the [Verify Disk structure](#) are stored in the log

Recover options

The log will monitor all options that have selected as part the appropriate recovery screen.

NTFS - these will include

- Cluster size
- \$MFT start cluster and start sector
- MFT count

Relative sector

FAT - these will include

- Cluster size
- Cluster 2 location
- FAT start
- Directory start

NTFS messages

The forensic report will do basic analysis on system files. This will include \$bitmap and \$logfile

Fixup error in MFT - and MFT has self checking built in, an error was found Cluster out of range - a cluster higher the length of the partition was requested

MFT entry not found at expected location - the specified location is not an MFT

MFT entry not found at cluster 4 - when an MFT is not set by BIOS, some values are tested

MFT entry not found at cluster 0xc0000- when an MFT is not set by BIOS, some values are tested

Cluster 0xc0000 has been set as start of MFT - a 'guessed' value has been used

Cluster 4 has been set as start of MFT - a 'guessed' value has been used

MFT for xxx not a directory MFT - a parent directory location is invalid

FAT messages

FAT parent directory not found for cluster : xxx - the parent directory was not found, dummy directory will be created. This is the directory that is pointed to by the '..' entry in the directory stub.

File truncated, found xxx expected yyy - the full file has been truncated, often due to a FAT entry indicating end of file.

Next cluster same as current cluster, incrementing value - this will cause the program to loop on single cluster, so the next sector will be selected automatically.

-0-

Keyword search

[Home](#)

The keyword search is part of commercial and forensic options. It allows files to be tested for keyword while being recovered.

#	Flag	Offset	Count	String	File name
0	---	1da220	1	y07	F:\test\backsew\ad\downloads\1-26-2009_018.jpg
1	---	69fe	1	for	F:\test\backsew\ad\downloads\1-26-2009_018.jpg
2	---	89fe	5	for	F:\test\backsew\ad\downloads\1-26-2009_018.jpg
3	---	8352	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
4	---	28ab50	2	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
5	---	25ca3	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
6	---	2294	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
7	---	8a70	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
8	---	941a	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
9	---	762f	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
a	---	4044	b	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
b	---	efc	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
c	---	9e703	b	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
d	---	8bc	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
e	---	b5f	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
f	---	76c1	2	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
10	---	c11	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
11	---	26bc	2	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
12	---	ae	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
13	---	c1f	8	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
14	---	983	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
15	---	1047	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
16	---	2795	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
17	---	2bec	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
18	---	17cd19	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
19	---	7b74c	3	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
1a	---	8910	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
1b	---	5287	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
1c	---	4e6b	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
1d	---	472	1	King Henry	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
1e	---	5296	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG
1f	---	7147	1	for	F:\test\backsew\ad\downloads\100_3657_100_3658.JPG

The log displays the results of a keyword search on the files. the fields are as described below

The original entry number

Flags

This is the type of search carried out, and can be any combination of the following

- c Character search
- u Unicode search
- i Case ignore

Offset

This is the offset of the first occurrence within the file

Count

This is the number of times the string has been found

String

The string being searched for

File

The name of the file the string was found in

-O-

Trace file

[Home](#)

Occasionally in software things go wrong. This is not intentional but with data recovery one is never in control of the data or media being read. There can also be problems with users not understanding the best procedure, or even pressing the wrong buttons.

The trace file is a simple text file (it stores no user data) but does track the user operations. It can therefore be very useful in isolating problems and giving a clear record of the 'buttons' pressed. With any program it can be difficult to remember exactly what menus were seen before a problem, or which options were selected..

The trace file is therefore a simple file to record user actions. There may be up to 9 such files stored in the log directory. A new file is started each time the program starts, and each numbered file is overwritten in sequence. If you do experience a problem, CnW support might request to see the latest such file. The contents may not make total sense to a user, but can be considerable help to CnW Recovery.

-0-

[Home](#)

Many software programs detect errors. Data recovery programs can be more prone to errors because they are dealing with hardware and file structures that are often corrupted, or in an unknown state. Therefore it is not impossible that the program may hang or crash. The following will give some guidance of how to handle such situations.

To help with 'remote' problem finding the program stores short [trace files](#). A new one is started everytime that CnW starts and has the name trace?.txt where ? is a number of 1 - 9. The files are created, and re-used in sequence so often the releavnt file will be the one with the most recent date. However, if the problem occurred a few runs before, then the trace for this run will still be available. Sending this file to CnW will assist in resolving any problems.

- [An error occurred in an unknown file](#)
- [Cannot read first mft, copy failed](#)
- [Missing files](#)
- [All files are only 4K, or 8K long](#)
- [Files not saved](#)

-0-

An error occurred in an unknown file

[Home](#)

This is a Microsoft error message that is very unhelpful, and at times can be a nightmare to track down. It normally relates to accessing a file on the system hard drive - ie not on the drive being recovered. It can be caused by an illegal filename, or sometimes due to a virus. Many virus checkers will delete a file when a virus has been detected which can mean that CnW software thinks there is a file which has actually been deleted. A quick solution at time could be to run the recovery program with the virus checking temporarily disabled. Becuase the recover process does not actually run a file, the virus will not run either - but a full virus check should be performed ASAP.

All such errors should be reported to CnW Recovery, where we will try and stop them occurring in later versions of the software.

-0-

Missing files

[Home](#)

There are several reasons for missing files and the following section will suggest approaches that can be tried to recover all files

The subdirectory for the files has become separated

If recovery is being performed using the system directory structure then a corrupt directory entry can cause all files under that directory to be lost. The solution for NTFS is to recover from File Entries. This may mean that files are stored in dummy directories, but they will be found. For a FAT disk, use the Recover from directory stubs option.

A complete partition is lost

If the partition table has been corrupted, then it is possible for a complete partition to be lost. The solution is to use the Partition function and Analyse Partitions, and then Scan for previous partitions.

On a Unix disk, different file systems may be installed, but typically a Unix disk does not always have a DOS compatible boot sector with partition tables. Again, a partition analysis and scan for previous partitions will find any existing partitions.

-0-

Error Messages

[Home](#)

The following is a list of error messages that may be displayed.

Disk parameters look dubious, run in manual mode

The wizard has done an analysis of a FAT disk and can not determine good values for cluster size, and cluster 2 start. It is advised to run in manual mode and determine working values.

No MFTs seen

There have been no MFTs detected from the partition. This is probably because the \$MFT file has been corrupted. The solution is to use the option to scan the drive for MFTs entries

Only unallocated data may be recovered from unknown media

The type of media has not been determined ie it does not now if it a FAT, NTFS etc disk and so only an Image Raw function can be used.

Very few files found, try an image scan?

The wizard has not found many files. Try an Image Raw scan to see if there are more files

ERROR_NOT_ENOUGH_MEMORY 8L

This is a Windows error message - sometime caused when there are a very large number of subdirectories or files produced.

Not enough server storage is available to process this command - 1130L

This message can occur when writing large files (typically disk images) to a disk drive when using XP. The solution is described in the following Microsoft knowledge base link.

<http://support.microsoft.com/kb/304101>

Cannot write to *** m_l0SError 0x**

This is an error detected when writing to a disk. It is possible to determine the error message from the number given, but feel free to contact CnW Recovery along with the error number for help

-0-

All files are short

[Home](#)

With FAT disks a common problem is for files to be limited in length. The length is always a value such as 4K, 8K, 16K. The most common reason for this is because the file allocation table has been corrupted or blanked. The cure though is simple in that the disk should be read using the Ignore FAT button. This will read the file sequentially for the correct length. The problem is that as the FAT has been lost, so have details of possible fragmented files. Heavily edited files, or long files may not read correctly and [data carving](#) may be necessary to regenerate a readable file.

Recovery Options FAT disks

☒ Full recover
☐ Recover from directory stubs
☐ Recover from FAT
☐ Scan disk to check for sector positions

☐ Overwrite existing files
☐ Recover deleted files
☐ Ignore FAT
☐ Recover unused space
☐ Recover slack space

Recover All
Select files
Cancel

Disk parameters

Cluster size: 20 Cluster 2 location: c0

FAT start: 32 FAT length: 3f

Dir Start: b0 Sector count: 7cfd

Use Fat 2: ☐

Enable: ☐ File Filter...

Analyse disk parameters

Scanning : 511900

Serial Number: 00000000

Display in Hex: ☒

Output directory: g:\#3\job15\ Browse...

-O-

Files not saved

[Home](#)

Occasionally files look as if they are being save but in fact are not saved. The following are possible explanations

Licence not applied

If a licence number has just been entered it may be necessary to restart the program. The top left of the CnW windows should indicate that it is a licenced, either full or 30 day version. The demo will not save any files

File filter set

If the file filter is set then only selected files will be copied. If nothing matches the requirement, no files will be saved. It must be noted that for data carving dates and file names are created on the fly, and may not match original details

30 day licence expired

At the end of a 30 day licence, the software reverts to demo mode. The licence can be updated to a full licence by purchasing a second 30 day licence. At this point CnW must be e-mailed with details in order to produce a full licence. The operation is not automatic.

-0-

CD Physical structures

[Home](#)

CDs are round and shiny, and although they all look the same, they come in several flavours, and several formats. The following chapters discuss the physical formats which have evolved over time and with enhanced drive capabilities. Relevant key terms are

- [Disk at once](#) (DAO)
- [Track at once](#) (TAO)
- [Session at once](#)
- [Packet writing](#)

The formats have evolved generally to allow smaller sections of the disk to be written at a time. This means that for multiple session disks, there is less wasted overhead when a session is added. The logical capacity of a disk can also vary. All this detail information can sometimes assist with a forensic investigation, so the CnW Recovery program will whenever possible determine, and log the details of the disk.

Other critical points to understand about disk is if a disk is open or closed. Other terms that will be discussed are Run in, Run out and TOC (table of contents). These are all parts of the physical recording of a disk that are normally only seen by the CD reader, and not the application reading the logical disk.

When it comes to RW disks, there are other variations possible.

Any modern CD-RW or DVD-RW will read all variations of disks, but there can be limitations.

Although logically similar many of the CD concepts are not part of DVDs.

-0-

Disk at once

[Home](#)

Disk at once recording will write a complete disk in one stage. By doing this, no space is wasted between tracks, but also all data has to be generated before writing to the disk starts. Thus lead in and Table of Contents must be known, along with every sector location that will be used. The safest way to generate this type of disk is to produce an image on the hard drive first, and then write the disk. The write laser is never turned off in this process.

Because of the fixed table of contents, only a single session may be written, and the disk is therefore closed at the end, and no further data may be appended. The process is very useful when producing a master disk that will be used for pressing. When the data is written, the laser is never turned off. It could be argued that it makes it a very reliable method of writing, but there is no flexibility about adding extra sessions.

Disk at Once is often used to produce master disks for mass production, and is typically used for audio disks.

When possible CnW Recovery software will log the disk mode in the status part of the log.

See also

[Track at once](#)

[Session at once](#)

[Packet writing](#)

-0-

Track at once

[Home](#)

CD Track at once, as the name implies, records just a single track, and then writes a run-out sequence on the disk. Before the next track is written, a run in sequence of 2 blocks is written. These run-in (of one link block and 4 further blocks) and run-out sectors are ignored by disk readers, but occasionally can be heard as a click by an audio player.

When writing, the laser is turned off between tracks. Upto 99 tracks may be written on any CD

-0-

Session at once

[Home](#)

Session at once is used by CD-Extra, and is a subset of DAO.

CD Session at once is very similar to Disk at Once, except further sessions can be added. With Disk at Once, the disk is finalised and no further data can be added. The boundaries of multi-session disks can be determined, and if required, each session can be read separately. Forensically, it is therefore possible to discover which files have been added and which removed, or edited between sessions.

[CnW recovery](#) routines will operate with all variations of CD recording

-0-

Packet writing

[Home](#)

Packet writing allows the smallest increment to be written to a disk.

A packet always has a 7 block overhead, 4 blocks run-in, 2 blocks for run-out and a link block. The packet can then be a fixed size, or a variable size. This compares to a track at once where the minimum track length is 300 blocks, with an extra 150 blocks for run-in, run-out and linking. Thus, there are virtually no spaces between sessions.

-0-

CD terms

[Home](#)

To understand CDs, it is necessary to know the elements that make up a CD, such as Lead In, TOC, PMA. This section will outline what each term actually means, and how the CD makes use of it. A curious aspect of CD-ROMs is how much of the formatting is done by the drive. On most PC media, there are just a number of sectors and the software controls all partitions etc logically. By comparison, the CD-ROM drive does most of the control, and often locks out areas of the disk.

Table of Contents (TOC)

The table of contents stores the start and length of each track on a CD. There is a maximum of 99 possible tracks.

For a finalised, or closed disk, the TOC is stored in the the Lead In area of the disk. For disks that have not been closed, or finalised, the track information is stored in an area known as Program Memory Area (PMA) which is only accessible with CD-RW drives. This is one reason that all data recovery of CDs is best performed using a RW drive, and not just a CD-R drive

Lead In

Lead in is written when a session is closed. It occupies about 9MB of space, and contains the TOC for the session. It also points to the next area on the disk that could be written to.

Lead Out

Lead out is written when a session is closed, and is written after the data area. The first lead out on a disk is about 13MBs, but subsequent ones are about 4MBs. Only CD-RW drives can read the data if the lead out is missing

Track

Session

A session is a sequence of lead-in, data, and lead-out. Drives have evolved to handle multi-sessions, so old drives may have problems with some multi-session disk, typically the issue will be that they will only read the first session.

Subcode Channels

Reading documentation on CDs, one will find reference to subcode channels, with names P,Q,R,S,T,U,V and W. These are one bit codes, that are added together to provide extra information, and is probably beyond the scope of this chapter.

Red Book

When the first CD was developed by Philips and Sony in 1980, the story goes that the documentation was kept in a binder with red covers. Ever since, the documentation has been known as the Red Book. Apart from recording of tracks etc, the Red Book also contains details of how audio files are sampled. All musical CDs conform to this standard, with sampling at 44.1KHz. It is interesting to note, that even on data CDs, sector locations are often described as by Frame, Second, Minute, where there are 75 frames a second.

Yellow Book

The Orange book in 1984 really describes the first data CDs. Evolution is always difficult, and obviously there were issues with backward compatibility. A significant step between CD-DA (Audio) and Data disks is the level of error correction. Audio disks do have error detection and correction using Cross Interleaved Reed Solomon codes. The data block was 98 x 24 bytes, or 2352 bytes. For data, an extra 288 bytes of error correction was added along with 12 sync bytes, and 4 header bytes. The remaining data length is 2048 bytes, a nice computer length, 4 times the length of normal disk sector.

Orange Book

This was published in 1988 - the enhancement here was CD-R, recordable CDs, and multi-session CDs. TAO was included as part of the enhanced spec. In 2000, the spec was enhanced again to allow for 80 minute CDs, rather than just 74 minute CDs.

-0-

-0-

[Home](#)

Glossary of terms

Cluster A group of sectors, and normally smallest amount of space a file can use, ie a 1 byte file will always take up 1 cluster of space on the hard drive.

Endian Little and Big endian define how a computer stores its numbers. Any number over 255 requires more than one byte to store, and little and big endian refer to the sequence. Little endian stores the lowest byte first, and big endian stores the highest byte first. So if we want to store the value 266, this is 10AH. In Little endian it would be stored 0A 01 and Big endian, it would be store 01 0A

Little endian is also known as Intel byte ordering, and Big Endian as Motorola byte ordering. It could also be PC and MAC)

FAT File allocation table - used on DOS/Windows systems to track which sectors have been written to

FAT12 12 bits per cluster, max 4096 clusters

FAT16 16 bits per cluster, max 65536 clusters

FAT32 32 bits per cluster, max 4G clusters

GUID Globally Universal ID

MFT Master File Table, file descriptions in NTFS. A two sector record that always starts with the text FILE

MD5 16 byte hash value

SHA-1 20 byte hash value

Slack Space at the end of a file, but contained within a cluster that has been used

Unallocated space Complete clusters that are not used by current files

-0-

Useful links

[Home](#)

How to delete file protected by Trusted Installer. These could be system device driver files that need updating.

<http://helpdeskgeek.com/windows-7/windows-7-how-to-delete-files-protected-by-trustedinstaller/>

-0-

Addresses, and contact details

[Home](#)

CnW Recovery software is a PC program designed to recover data from all PC media that has been corrupted, partially formatted, or partially overwritten. The links below will take you to the relevant pages on the main [Website](#) www.cnwrecovery.com for downloads etc

For any questions, ideas, please email info@cnwrecovery.com

Any problems, or errors, please report by e-mail and they will be investigated as soon as possible.

Postal address

CnW Recovery Developments Ltd
14 King Henry's Road
Lewes
East Sussex
BN7 1BT
UK

CnW Recovery Program Download - [free demo](#) that is described by this manual

-0-

Index

- 3 -

3GP and MP4 recovery wizard 49

- A -

Addresses, and contact details 322
All files are short 311
Alternate Data Stream 149
An error occurred in an unknown file
Apple Volume Header 279
AVCHD reconstruction 287
AVCHD recovery 47
AVI Recovery 68

- B -

Basic Rules of Data Recovery 35
BIOS Parameter Block BPB 270
BIOS Parameter FDC descriptor for FAT 118
BIOS Parameter FDC descriptor for NTFS 140

- C -

Camcorder Recovery 108
Cannot read first mft, copy failed 146
CD Recovery 98
CD terms 318
CnW Recovery forensic investigation tools 201
Configuration for CnW Recovery Software 26
Create new video DVD 45

- D -

Data Carving 215
Data Carving with an Excel File 218
Date selection 235
Deleted file recovery 96
Deleted Partition 153
Demo program 14
Demo Status 15
Directories 32
Directory selection 236
Discover deleted files 204
Disk at once 313
Disk clusters 277
Disk imaging 76
Disk scan 228
Disks with single head failure 82
Dongle installation 19

DVD Properties 213

- E -

eMail Extraction 285
eMail restoration 290
Error Messages 310
exFAT 126
Extract and join 288

- F -

Failing disk drive 64
Fake memory test 289
FAT 32 deleted file recovery 122
FAT directory entry 273
FAT Disk restore 112
FAT File allocation table validation and correction 124
File details 293
File extension selection 234
File fragments 298
File selection based on MD5 value 238
File size selection 240
File Validation 221
Files lost when NTFS reloaded 144
Files not saved 312
Forensic analysis tips 231
Formatted disk recovery 65
Fragmented 3GP/MP4 files 257
Fragmented AVI files 173
Fragmented file processing 167
Fragmented Files 171
Fragmented Zip and DOCX files 261

- G -

General NTFS Recovery 175
Getting started - General data recovery 70
GoPro video recovery 193
GUID Partition sectors 267
GUID Partition tables 94

- H -

Hardware config 34
Hardware failure - what next? 37
How to find and recover lost files 154
How to recognise type of CD/DVD 100
How to recover corrupted partitions 92
How to recover FAT disk when boot sector is missing 121
How to recovery FAT disk when boot sector and one FAT is missing 116
How to use incremental imaging to recover damaged drives 80
HP Mediavault recovery 254

- I -

Image file selection 83

Imaging failing drive 181
Import List 242
Installation 17
Introduction 10
ISO9660 and Joliet investigation 206

- J -

Job details 299
Jpeg images and metadata 170

- K -

Keyword search 304

- L -

Linux and Unix recovery 127
Log overview 291

- M -

Macintosh Recovery 130
Magnetic Media Recognition 95
Manual Data Carving 217
Master Boot Record 265
Media detection 22
Merge disk images 284
MFT Parse 211
Mini DVD recovery 43
Missing directories and files on a FAT disk 120
Missing files 309
MP4 disk layouts 186
MP4 file structure 191
mp4_scan 190
MTF .BKF files 133
Multi-session UDF 107

- N -

NSRL Hash tables 226
NTFS directory entry, MFT 275
NTFS forensic investigation 209
NTFS MTF range 141
NTFS Recovery 134
NTFS with confused partitions 147

- O -

Overview 232

- P -

Packet writing 317

Partition analysis mode 88
Partition Table structure 90
Partitioned disk recovery 67
Partitions, analysis and recovery 85
Photo recovery 180, 46
Physical Media Test 62

- R -

RAID boxes and configurations 249
RAID configuration 247
Raid disks 78
RAID drive selection 245
RAID JBOD 251
Raw files 161
Rebuild video disk files 110
Recover FAT32 disk when it has been reformatted as NTFS 125
Recover video from camcorder with a hard drive 178
Recovering files from image format 165
Recovering when a new /different operating has been loaded 151
Recovery from a drive with many bad sectors 156
Recovery of lost files on an otherwise working disk 179
Recovery options 28
Registration 30

- S -

Search for MFTs 142
Search for sector 297
Search for strings 222
Search String 163
Session at once 316
Software limitations 16
Split directories 283

- T -

Terms 320
Trace file 306
Track at once 315
Typical 3GP corruptions 260
Typical data recovery procedures 73
Typical RAID setup parameters 253

- U -

UDF Anchor Volume 103
UDF forensic investigation 207
Unerase CD-RW 104
Unrecognised media 42
Useful links 321
User passwords

- V -

Verify disk structure 59
Video recovery from memory devices 185

Video recovery from mini-DVDs 183
Video scan of hard drive 58
View sector on hard drive, flash memory or CD 74
Virtual disk image 230
VMFS sectors 281

- W -

What to do when media has failed 35

- X -

XML Forensic Report 223

- Z -

ZIP and DOCX recovery wizard 56

