

Computer Forensics: Final Report

Thomas Partington, 1st May 2007

Supervisor: Mike Stannett. Module: COM3020

This report is submitted in partial fulfilment of the requirement for the degree of Bachelor of Engineering with Honours in Software Engineering by Thomas Partington.

All sentences or passages quoted in this report from other people's work have been specifically acknowledged by clear cross-referencing to author, work and page(s). Any illustrations which are not the work of the author of this report have been used with the explicit permission of the originator and are specifically acknowledged. I understand that failure to do this amounts to plagiarism and will be considered grounds for failure in this project and the degree examination as a whole.

¹[I have lined up your signature, the date (now generated by MyDate), etc, using tabular.]

Name: Thomas Partington

Signature:

Date: 1st May 2007

¹MPS changed:

Contents

1	Introduction	1
2	Literature Survey	2
2.1	Hidden Data - Where to look	2
2.2	Layer 1 – Raw Binary Data on a Storage Device (Hard Disk)	3
2.2.1	Bad Sectors/Tracks	3
2.2.2	Finding Data Hidden in Bad Sectors/Tracks	4
2.2.3	Host Protected Areas and Device Configuration Overlays	4
2.2.4	Hard Disk Imaging	5
2.3	Layer 2.1 – Volumes/Partitions	6
2.3.1	Definitions	6
2.3.2	DOS Partitions	6
2.3.3	Volumes/Partitions – Hidden Data	8
2.4	Layer 2.2 – File Systems	9
2.4.1	NTFS – New Technologies File System	9
2.4.2	Data Obfuscation: Deleted Data and Possible Recovery	10
2.4.3	Zero-Footprinting: Wiping the evidence	11
2.4.4	Hidden Data	12
2.4.5	Encrypted Data	13
2.5	Layer 3: Data In Context (Files)	14
2.5.1	Event Reconstruction	14
2.5.2	Hidden Data: Steganography	15
2.6	Current Computer Forensics Tools	17
2.6.1	EnCase by Guidance Software	17
2.6.2	Forensic Toolkit by Access Data	18
2.6.3	ProDiscover Forensics	18
2.6.4	SMART by ASR Data	18
3	Requirements and Analysis	18
3.1	The Tools	19
3.1.1	Analysing a Hard Disk and Creating an Image	19
3.1.2	Partition Analysis	21
3.1.3	File System Analysis	22
3.1.4	Encrypted Files	23
3.1.5	Event Reconstruction	24
3.1.6	Steganalysis	24
3.1.7	General Requirements	25
3.1.8	Non-Functional Requirements	25
3.2	Specific Requirements	25
4	Design	29
4.1	X-Machines	29
4.2	Partition Analysis Tool	38
4.3	File System Analysis Tool	38
5	Implementation	43
5.1	Disk Imaging Tool	44
5.2	Partition Analysis Tool	46
5.3	File System Analysis Tool	47
6	Testing	48
6.1	Disk Imaging Tool	48

6.1.1	Test Specification	48
6.1.2	Test Frames	50
6.1.3	Results	51
6.2	Partition Analysis Tool	53
6.2.1	Test Specification	53
6.2.2	Test Frames	54
6.2.3	Results	54
6.3	File System Analysis Tool	55
6.3.1	Test Specification	55
6.3.2	Test Frames	58
6.3.3	Results	58
7	Conclusions	59
7.1	Summary	59
7.2	Work Achieved	60
7.3	Suggestions for Future Work	60
8	Appendix A	i
8.1	X-Machines	i
8.2	Screen-shots of the Prototype System	v
9	Appendix B	xiv
9.1	Test Frames	xiv
9.2	Test Results	lx

Abstract

With 95% of the world's documents being created by using a computer [Bal04], hard disk capacities increasing to 1TB and the average computer becoming smarter - the demand for more advanced/efficient computer forensics techniques is on the rise. Criminals are finding new ways to hide or eradicate evidence of their computer activities and it's important that computer forensics analysts are able to keep up.

The aim of this project is to create a suite of programs which implement and automate computer forensics techniques in order to locate and analyze digital evidence.

To date I have discovered many techniques criminals can use to hide and delete data. These techniques range from the simple (e.g simply hitting the delete key), to the more advanced (e.g corrupting partition tables, altering the file system, moving data to normally inaccessible areas of the disk), to the complex (e.g advanced steganographic techniques).

1 Introduction

A computer can be involved in a crime in a number of ways [Vac05]:

- It can be the target of the crime, e.g if it is hacked into.
- It can be the instrument of the crime, e.g using it to commit online fraud.
- It can be the evidence repository for other crimes, e.g storing illegal images.

In all of the above cases the computer would need to be investigated for evidence using established methods which may vary depending on the crime being investigated. The collection, preservation, analysis and presentation of computer-related evidence that is sufficiently reliable to stand up in court is called *computer forensics* [Vac05]. *Evidence* in the context of computer forensics is data collected from a computer which may prove that a crime was committed and/or may prove the events leading up to a crime.

Computer crime is on the rise, although it is difficult to find figures, and criminals who use computers are finding new methods of covering their tracks. These methods usually involve destroying evidence on the computer, similar to how someone might shred a paper document, or hiding the existence of the evidence in a similar way that a criminal might clean a crime scene. Sometimes a criminal may even plant evidence on the computer in an attempt to mislead an investigator.

Computer forensics examiners use a range of specialist software tools in order to preserve, locate and recover evidence quickly. However in order for the evidence to be reliable in court the investigator needs to have a knowledge of the computer forensics theory of how and why it was possible to recover the evidence as courts in the US ruled that the software packages used can't be considered 'experts' [MR04]. Also the tools currently used are unable of finding some hidden evidence such as data placed in a Device Configuration Overlay or embedded in another file.

The aims of this project are to research the computer forensics techniques currently being used by professionals and to develop a suite of automated tools capable of collecting evidence from a computer in such a manner that they would be reliable in court.

Section 2 contains a review of relevant literature describing methods used by criminals to hide evidence along with the techniques used by computer forensics experts to find the evidence. Section 3 analyses and describes the requirements of the project in detail and section 4 contains the design of the system developed. Section 5 describes problems encountered during the implementation and explains how I overcame these problems. Section 6 explains how the system was tested and the results of the testing process. Section 7 contains the conclusions reached from this project, the work achieved and suggested work for future projects. Appendix A contains analysis and design diagrams not included in the main document and Appendix B contains test data omitted from the main document due to space limitations.

2 Literature Survey

2.1 Hidden Data - Where to look

In order to analyse the large amounts of data found in modern digital systems it is necessary to identify system layers where *evidence* could reside. These layers are called layers of abstraction and can be defined as a function of inputs and outputs[Car03]. These functions define how to interpret raw binary data (low layers) into higher level structures such as partitions and files (higher layers). This allows an investigator to identify expected data which can then be ignored when searching for evidence. Several Computer Forensics experts have identified different layers of abstraction:

Pollitt and Hama [HP98] identify 3 basic layers. The lowest of these is the actual physical storage device (hard disk, RAM, USB memory device etc.). This layer can contain hidden data along with the other data on the disk. It is necessary for an investigator to make a copy of all the data on the disk in order to authenticate any evidence found. This² process of duplicating the data is called disk imaging and will be discussed later. The second layer comprises the logical organisation of the data into volumes/partitions and file systems. The third layer is the “data in context” which contains the user and system files, blocks and allocation units which will be discussed in a later section. A diagram of these layers is shown in figure 1

Figure 1: Layers of Abstraction identified by Pollitt and Gord

Carrier [Car05] identifies further layers, not restricted to the hard disk, which can contain useful evidence. The first layer contains both the physical storage device and a network, which can be thought of as a network of storage devices. Volumes/partitions are considered to constitute a layer in their own right as they need to be analysed to identify and determine the layout of file systems. Carrier also includes the system’s memory (RAM) in this layer. The third layer contains the file systems, which are located in the partitions, swap space (this is used by memory and will be discussed later) and database systems. Blocks and allocation units are part of a file system’s structure and are therefore included in this layer. The fourth and final layer is the application layer which contains user and system files.

To keep things simple I will divide the discussion of computer forensics and data hiding/destruction techniques into sections related to the layers of investigation identified by Pollitt and Hama. Although it is theoretically possible for a criminal to use other techniques which may not leave a footprint however this is unlikely if the evidence has been stored on the hard disk at some point.

²MPS changed: is

2.2 Layer 1 – Raw Binary Data on a Storage Device (Hard Disk)

A hard disk is made up of a number of rigid platters, each of which is coated on both sides with a magnetic material. A bit is stored on the disk by placing an electromagnetic charge at the appropriate location. There are various encoding methods for storing bits which do not need to be discussed here. The disk is organised into small areas called sectors and it is these sectors which are addressed by the operating system (OS).

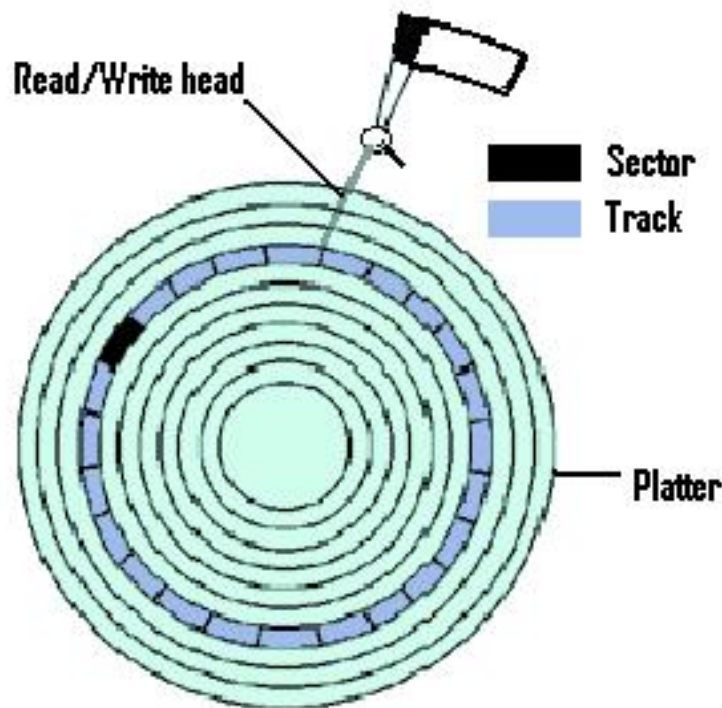


Figure 2: The layout of a hard disk [img]

2.2.1 Bad Sectors/Tracks

The following is based on a comprehensive survey by Sammes and Jenkinson [SJ00]. It is possible for disk sectors to contain micro-defects which makes them unusable. Sammes and Jenkinson identify strategies used by the disk controller to overcome the effects of these ‘bad’ sectors and they identify ways these strategies can also be used to hide data:

When one or two consecutive bad sectors are found the controller can adjust the sector formatting so that the defect occurs between sectors or between tracks; it would then be as though this defect doesn’t exist. Whilst it is technically possible for a criminal to perform his/her own low level formatting it would require a great deal of expertise to do so and still preserve the bad sector mapping on some hard disks; on other disks it essentially performs the same function as a high level format except the data in all sectors is scrubbed. Therefore this method of data hiding isn’t likely to be encountered during an investigation.

If many bad sectors are found on a track – a track is a ring of sectors around the hard disk

platter (see figure 2) – the controller can mark the entire track as bad and then assign an alternative track to replace it, the replacement track is normally located at the end of the disk. When the OS attempts to access a sector in the bad track the disk controller will look up the address of the alternative track and access it instead. Using the appropriate disk controller commands it is possible for a user to manually assign alternative tracks. A criminal could place data he doesn't want investigators to find in one track of the disk and then mark that track as bad and assign an alternative track containing legitimate data. It would then be impossible for anyone to access or even be aware of the hidden data by using the OS.

A simpler procedure would be for the criminal just to mark the track as bad as this would still render the data inaccessible but it would have the apparent effect of creating a hole in the disk.

2.2.2 Finding Data Hidden in Bad Sectors/Tracks

In the case of a track or sector simply being marked as bad and not being reassigned I have already mentioned that there would be an apparent hole in the disk. It is possible to use the disk controller to determine the number of formatted tracks, the actual number of available tracks is often written on a label on the disk casing, if the number of formatted tracks is less than the expected number of tracks then there are some tracks which are marked as bad. However this is not necessarily evidence of intentional data hiding, since defects can occur on the disk, and it is possible that the track was marked as bad because of actual faults within it. The disk manufacturer often creates a list of bad sectors which is written on the disk casing [SJ00], and this list should be checked against the list of bad sectors produced during analysis to determine which are genuine and which may contain evidence. All the bad sectors which may contain evidence should be analyzed.

During the low level formatting process each sector has a number of fields associated with it. One of these fields is the ID field which normally contains the address of the sector. If a track has been reassigned to cover the hole left by a bad track then it will have an appropriate flag set in the ID field for each sector in the track, which is located before the data area of the sector. Its address will also be changed to the address of the sector it is replacing, and this address is also stored in the ID field [SJ00]. Using the known disk geometry it is possible to locate the supposedly bad sector and mark it as good, and the actual address of the replacing sector can be calculated and reset, as can the flag identifying it as reassigned. The sectors can then be analysed in the same as all the other sectors on the disk. Alternatively the ATA SMART Command **SMART READ DATA** (available on virtually all ATA hard disks) can be used and the **Reallocated Sectors Count** attribute will identify if any sectors have been reallocated [Vid05].

In the event that the suspect used low level formatting to readjust the sector formatting then it will be necessary to analyse the system for evidence of formatting tools and their use. Event reconstruction would be needed to determine how the tool was used so that its effects could be reversed, this would be extremely complicated and would require the use of advanced data recovery techniques. This is complicated because it would need to be determined how the programs used by the user actually work and most basic event reconstruction methods don't provide this data. These methods will be discussed later.

2.2.3 Host Protected Areas and Device Configuration Overlays

The *Host Protected Area* (HPA) is a reserved area on a hard drive [GHR06]. Data hidden in this area is not accessible via the OS and it can require special tools to detect and remove the area (leaving the data intact). Hard disk controller commands can be used to organise the disk into user-accessible and protected areas; the protected areas are located after the user-accessible area - this is illustrated in Figure 3.

It can be determined if the drive in question has a HPA set by using the disk controller command **READ NATIVE MAX ADDRESS** to determine the original size of the disk, unless the disk supports 48-bit addressing in which case the command is **READ NATIVE MAX ADDRESS EXT**. The value returned, which is the maximum sector address, can then be compared to the address of the maximum user addressable sector given by the command **IDENTIFY DEVICE**, if the values are

Figure 3: A hard disk with a 2GB Host-Protected Area

different then a HPA exists [Vid05]. Note - These commands are used by all ATA hard disks, it will be important to identify commands that can be used by other types of hard drive (e.g SCSI). The detection of HPAs is important when it comes to hard disk imaging discussed in section 2.2.4.

For example a criminal could hide data on a 60GB HDD by making sure the data was originally written in the last 1GB of the disk and then setting a HPA which covers the last 1GB of the disk. This is done by setting the maximum user address to be at the 59GB mark. The data would now be inaccessible until the HPA was removed (i.e the maximum address reset to 60GB).

A *Device Configuration Overlay* (DCO) allows a user to limit the capabilities of a hard disk by disabling features and/or reducing the capacity of the disk. This is done via the disk controller command `DEVICE CONFIGURATION SET` [GHR06]. If a DCO exists then when using the `IDENTIFY DEVICE` to find the capacity of the disk the capacity of the DCO will be returned [Car05]. To remove the DCO the command `DEVICE CONFIGURATION RESTORE` is used although this won't remove a HPA [GHR06]. Detection and removal of DCOs is necessary during hard disk imaging which is discussed in section 2.2.4.³**[HARD-CODED CROSS-REFERENCE!!!!!!]**

For example a criminal could hide evidence on a 80GB HDD by making sure the data was originally written to the end of the disk (e.g within the last 2 GB) and then setting the capacity of the disk to be 78GB using a DCO. The hidden data would then be inaccessible until the DCO was removed. A criminal could also use an HPA and a DCO in conjunction to create a fake *stopping condition*. For example when retrieving the true capacity of the disk to find a HPA the value returned wouldn't include the DCO which the investigator might then overlook.

There are few, if any, currently available computer forensics tools which are capable of detecting both HPAs and DCOs [GHR06, Vid05]

2.2.4 Hard Disk Imaging

In order for any digital evidence to be admissible in a court of law an investigator needs to prove that the evidence hasn't been contaminated or destroyed. A common way of doing this is to create a bit-stream image of the original storage device and then confirm that the copy is identical by using a checksum algorithm such as *MD5* [MR04]. The checksum algorithm generates one value based on the input from the copy, and another value based on the input from the device being copied. The chances of two arbitrary files having the same checksum are sufficiently small that if the two values match then the copy is considered to be identical to the original.

The image can either be written to a file or copied straight onto another hard disk. However Carrier observes that if the image is written straight to disk it can be difficult to identify where the image begins and ends [Car05]. This means it is possible that deleted data from a previous case could contaminate the new evidence and although computer forensics expert use disk wiping tools between cases these aren't always reliable as I will discuss later. The data could be safely written to a brand new hard disk, however law enforcement agencies who investigate thousands of cases may be unwilling to invest in so many new disks.

It is also important to make sure that no data is altered on the original drive during the imaging

³MPS changed:

process – *hardware write blockers* are used for this. A hardware write blocker is a physical device which is connected between a disk and the disk controller. It has the following properties [Vid05]:

- It won't permit any modifying operation to be transmitted to the protected device;
- It returns all data requested by a **read** operation;
- Any error condition reported by the storage device is reported to the host;

During the imaging process it is important for the tool to log any operations performed and any errors reported along with the appropriate date/time stamps (to maintain the chain of custody). If the imaging tool is unable to read certain sectors for whatever reason then it is accepted practice for the tool to write 0 s and make an appropriate entry in the log [Car05]. Whilst this means that some bad sectors won't be copied and examined, the data in those sectors couldn't be read anyway and so were unlikely to be used by criminals for storing data.

This imaging process described relies on the suspect hard disk being switched off when it is originally encountered. If it is switched on then it shouldn't be shut down as this could lead to vital evidence being erased. Methods for live image acquisition should be used, and there is the advantage that data from memory (RAM) can also be recovered. Due to time and space limitations I will only deal with acquisition of data from hard disks that were switched off (*dead analysis*).

Once the disk has been copied all further analysis uses the copy, which will be considered by the courts to be original [MR04].

2.3 Layer 2.1 – Volumes/Partitions

A hard disk is typically organised into a number of smaller, more manageable, sections called volumes and partitions. Each of these sections may contain its own file system and/or operating system. In this section I will describe the basic fundamentals of volumes and partitions, using the MS-DOS partitioning system as an example, so that it is possible to locate file systems and to highlight possible hiding places for data.

2.3.1 Definitions

Wikipedia defines a volume as “a single accessible storage area with a single filesystem, typically (though not necessarily) resident on a single partition of a hard disk” [Wikb]. The volumes normally encountered during computer forensics investigations will most likely be hard drives, although it is important to note that the drive itself isn't a volume – the volume comprises the addressable sectors on the disk and will probably cover the entire disk capacity unless an HPA or DCO is present.

Wikipedia defines hard disk partitioning as “the creation of logical divisions upon a hard disk that allows one to apply operating system-specific logical formatting” [Wika]. A more helpful definition comes from Carrier; “A partition is a collection of consecutive sectors in a volume” [Car05]. The difference between this and volumes is that sectors in a volume need not be consecutive.

Partitions are contained within the volumes, this structure is shown below:

2.3.2 DOS Partitions

I will use the DOS partitioning system as an example because it is used by every version of Windows and by many Unix distributions but the basic ideas are shared by virtually all partitioning systems.

The layout of the partitions is stored in the first sector(s) of the volume in the partition table record or Master Boot Record(MBR). The first 446 bytes of the MBR are reserved for code used to boot the system and the remaining bytes contain the partition table [HIW]. The partition table contains 4 entries (DOS originally only allowed 4 partitions called primary partitions). Each entry in the table stores the partition's; starting address, ending address, size in sectors, type and

Figure 4: A hard disk with 6 partitions and a HPA

flags which indicate if the partition is bootable (only one partition may be marked as bootable) [Bro]. Although multi-boot systems exist there is usually code placed in one partition which will then jump to the appropriate boot code in another partition depending on the user input [Car05]. Technically only the partition containing the initial code with the jump statements is bootable.

If a user desires more than 4 partitions then the fourth partition will be made into a primary extended partition which contains a secondary partition. A secondary partition can contain a secondary filesystem (just like a normal partition) or it can contain both a secondary filesystem and a secondary extended partition. Secondary extended partitions can contain the same as a secondary partition. The secondary (and secondary extended) partitions also contain partition tables which point to the next secondary file system and secondary extended partition (if applicable). Thus secondary partitions are stored in a linked list structure as shown below:

Figure 5: A system with 6 Dos Partitions [Car05]

2.3.3 Volumes/Partitions – Hidden Data

It is possible to declare one or more entire partitions as hidden by setting the type of each partition you want to hide to an appropriate value, e.g 0x11. This type is stored in the partition table [Bro]. These partitions can be easily found by processing the partition table.

Alternatively a criminal could wipe or corrupt the partition table in an attempt to destroy evidence of files. Secondary Extended partitions have unique header values, or signatures, which identify them. An investigator could search for these signatures to try and locate some of the partitions. Of course, a criminal with knowledge of partitioning systems could also delete these partition tables. An alternative for the investigator would be to search for filesystems (which also have signatures) and determine the size of each partition from the data they contains [Car05].

There is no requirement for the data in a volume to fill all the space which is assigned to it. Any spare sectors at the end of the volume wouldn't be accessible by the OS, these spare sectors are called volume slack [BHS].

Figure 6: An illustration of volume slack [BHS]

Any spare sectors contained in the volume but not contained within a partition are known as partition slack [BHS]. Slack can legally occur both between partitions and at the end of the volume [Car05].

Figure 7: Possible locations of partition slack in a volume [Car05]

To determine if there is any slack space an investigator would analyse the partition tables to map the partitions. The total number of sectors in the partitions would be compared to the total number of sectors in the volume and if the values don't match then there is some partition slack.

If the system has fewer than 4 partitions then not all the partition table entries are used and a criminal could place evidence in the unused areas of the partition table – this data wouldn't be

accessed by the OS. To create more hidden space the criminal could create many partitions and place the data inside them and then delete the partitions (this wouldn't delete the data placed there). Systems which have 64-bit hardware use GUID Partition Tables (the details of which aren't important for this discussion) which support up to 128 partitions. Many of these entries in these partition tables may not be used, leaving quite a large amount of space which can be used to hide data [Car05]. Once again the partition tables would need to be located and analysed to determine which entries aren't used, these unused entries should then be checked to see if they contain all 0s – if they don't then they may contain hidden data.

Finally, the beginning of each partition contains sectors reserved for boot code. If the partition isn't bootable then this space isn't needed but it isn't accessed by the OS, it could therefore be used to hide data. An investigator should process the partition table(s) to determine which partitions are bootable. The reserved areas at the beginning of non-bootable partitions should be examined for hidden data [Car05].

2.4 Layer 2.2 – File Systems

File systems are used for the storage and retrieval of files and their corresponding details (e.g last accessed times, permissions etc.). A file system is located inside a partition, systems with many partitions may also have many different file systems. In this section I will discuss the principles of how file systems work, using NTFS as an example (as it is the default file system for modern versions of windows), how files are created and deleted and the strategies that can be used to recover hidden and deleted files.

2.4.1 NTFS – New Technologies File System

NTFS is an object oriented file system containing 3 major areas [Car02]:

- The Boot Sector;
- The Master File Table (MFT);
- The Data Area;

The Boot Sector is located in the first sector of the file system and contains boot code along with information about the file system. This information includes the size of the file system (in sectors) and the address of the first MFT entry [Car05].

For efficiency NTFS addresses larger data units called clusters rather than individual sectors. A cluster is a fixed even number of sectors, cluster sizes may vary between file systems and operating systems, a typical size for a Windows 2000 system using NTFS is 4096 bytes (usually 8 sectors) [Kue02]. The cluster size for the NTFS file system is also stored in the boot sector. The address of the first MFT entry given in the boot sector is the cluster address rather than the sector address.

A major feature of NTFS is that everything is treated as a file [Car05], even the MFT, and these files can be located anywhere in the file system with the exception that the first clusters reserved for the boot sector. This area where the files can be stored is called the data area.

These files are organised and managed by the MFT. Each MFT entry, also known as a file record, stores the metadata (access times, permissions etc.) for the associated file along with the file's content. These data are themselves stored in file record attributes, whose names always begin with a "\$", which have specific purposes, for example the \$STANDARD_INFORMATION attribute contains the last access times whereas the \$DATA attribute contains the file's content. The \$BITMAP attribute is used to store which file records are allocated and which are unallocated. Some file records are reserved for specific purposes, e.g. file record 5 is allocated to the root directory . File record 0 is allocated to the MFT and it stores data about the size and layout of the MFT, file record 1 is a copy of file record 0. File record 3 is allocated to the Bitmap file which is used to store allocation status of each cluster in the file system, if the value for a cluster is set to 1 then the cluster is allocated, if the value for a cluster is set to 0 it is unallocated [Car02], [Car05].

A file record attribute has its own unique attribute header which contains the type of the attribute (Standard Information, Data etc.), its name and a boolean flag which indicates where the attribute content is stored. If the attribute content is small enough to fit in the file record then it is resident and the attribute header will contain a pointer to the content [Kue02]. If the attribute content is too large to fit in the file record then it is non-resident and the attribute contains the addresses of the clusters which contain the content. These cluster addresses are stored as cluster runs; a cluster run is a number of consecutive clusters denoted by the address of the first cluster and the length of the run [Car05].

A file record is given a sequence number which is initially set to 0 when the file system is first created and every time the file record is allocated to a file the sequence number is incremented by 1. Every data structure (larger than 1 sector) in NTFS is given a fixup value which is incremented every time the system writes to that sector. Corrupted or partially written files will have different fixup values so they can be easily identified. Each file record has its own header which contains a signature value, its sequence number, the fixup values for the file and a pointer to the first attribute [Car05].

Figure 8: An NTFS file record with 3 attributes [Car05]

The only other detail of NTFS which needs to be discussed is the use of indexing. Indexing is done by use of a B-Tree for each directory where each node in the tree represents the files and directories located in that directory. The tree and its nodes are stored in either \$INDEX_ROOT or \$INDEX_ALLOCATION attributes. These indexing structures will be referred to in later discussions about finding files.

2.4.2 Data Obfuscation: Deleted Data and Possible Recovery

“Data Obfuscation refers to any attempt to cover a hacker’s tracks by deleting or otherwise destroying digital evidence” [Sar06]. When a file is deleted in NTFS the index node for that file is deleted from the tree and the tree is then reorganised causing the index to be overwritten, the file record(s) associated with the file is set as unallocated by changing the data in the \$BITMAP attribute of the MFT file record. The clusters which contain the file are set to be unallocated by changing the cluster values in the Bitmap file record to 0. The file’s contents and file record remain intact on the disk until they are overwritten by a new file being allocated to those clusters/file records.

A computer forensics expert can take advantage of this knowledge to find data a criminal thought was deleted. The most simple case would be when none of the data has been overwritten.

In this case an investigator would process the MFT looking for unallocated entries which would be analysed to find the location of their content which could then be extracted. A slightly more complicated situation would be if the file record has been allocated to another file and has been overwritten as it would no longer contain the address of the deleted file's content. Many files have a sequence of bytes at the beginning of the file that specifically indicate the type of the file [SJ00], these bytes are known as the file's signature or magic number. If an investigator knows what type of file he/she is searching for then they can search the unallocated clusters for occurrences of the appropriate sequence of bytes. However the sequence of bytes may occur coincidentally in the system and may not indicate the start of a file, these false results need to be filtered out either by the system or the user checking for the presence of other expected, or unexpected, values. This type of searching is called *data carving*.

There is a possible source of confusion when file records have been reallocated numerous times. If the clusters for a file are deleted and then allocated to a file which is deleted and then the file record for the second deleted file is allocated to another file there will be no obvious way of knowing which file record the clusters were associated with. A diagram of this situation is shown below:

Figure 9: A possible source of confusion with reallocated file records [Car05]

If some of the clusters allocated to a deleted file are overwritten then file signature searching probably won't work, unless the only cluster deleted was located at the end of the file. In this case advanced data carving techniques are required. One method is to use a process of elimination to identify which sectors *don't* belong to the deleted files, sectors that are identified as definitely belonging to the deleted files are bookmarked and are ignored by subsequent carving processes [Dic06]. The specification of the required file format (e.g .zip) is used to identify which data values are expected and at which locations, this is easy for text files as they store large amounts of ascii/unicode displayable characters. Dickerman gives examples of values expected for various file formats [Dic06].

However I haven't found any papers that describe methods of reconstructing files using the fragments that are recovered. For text files it should be possible to use probabilistic text processing methods to predict, with a reasonable degree of accuracy, which pieces of text belong to which document. For images it would be more difficult. It may be possible to analyse the file fragments and identify the bytes which represent the pixels at the edges of the fragments. It may be reasonable to assume that fragments which have the same (or similar) edge values can be pieced together like a jigsaw, although it would be necessary to account for slight variations in colour intensity between the pixels. One aim of this project will be to investigate these ideas further.

2.4.3 Zero-Footprinting: Wiping the evidence

Criminals, and legitimate computer users, are becoming increasingly aware of investigators abilities to recover deleted data and are therefore finding new ways to obfuscate data. One approach being

used is to modify the file system to try and mislead an examiner, for example a hacker may leave a message behind crediting a different hacker for the intrusion [Sar06]. Alternatively a criminal may attempt to remove the data left behind when a file is deleted.

“Zero-Footprinting refers to a program that is used to clean areas of the disk in order to completely destroy the original contents.” [Sar06].

Data wiping (or secure deletion) programs are becoming widely available and are being used more and more to destroy evidence left behind on a hard disk even when the associated file is deleted. These programs operate by overwriting the appropriate areas of the disk with random binary data, according to the US DoD the data is only securely deleted when it is overwritten by a sequence of similar bits (e.g all 1s) then their complements (e.g all 0s) and finally by random data [JR91].

It is likely that computer forensics experts will encounter data wiping projects more often as Windows XP comes with a built in scrubbing feature. However it has been described as difficult to use, too difficult for the average computer user, and no better than an average wiping program [SK01]. The problem still remains of how to recover evidence that has been wiped.

Data wiping software, like any other software, isn’t perfect. Its possible for the programs to leave some data behind which was meant to be overwritten. Researchers have tested some of the available data wiping tools and identified areas of failure (some of these areas will be explained in section 2.4.4) [Gei05]:

- incomplete wiping of unallocated space;
- failure to wipe targeted system files;
- overlooked registry records;
- system restore points;
- special file system structures;
- activity logs;

If the system requires more memory than is available it will use some of the hard disk space as an extension to RAM, this additional memory is called virtual memory and the file on disk which it uses is called the swap file or the page file. It is possible for this swap file to remain after the system is shut down and it could contain any sort of data that had been stored in memory. Wiping software may fail to erase this file and an investigator could use it to find evidence [Kue02].

Courts in the US have punished users of these wiping programs as they ruled that their use implies intent to conceal evidence [Gei05]. An investigator should therefore identify any evidence of their use. These tools leave behind distinctive features (like digital fingerprints) [Gei05].

Other researchers have analysed overwritten areas of a hard disk by using a magnetic force scanning tunneling microscope. They observed that due to mechanical faults the read head of a hard disk may not be perfectly aligned with the data that is to be overwritten (tracking misregistrations) causing some overwritten data to remain and be detectable [GBA⁺93].

Experts refer to this phenomena as Shadow Data and observe that it occurs either because of variations in signal strength from the write head as it passes over the disk or due to mechanical faults which cause the write head to wobble in an s shape [Kue02].

2.4.4 Hidden Data

A file stored in an NTFS system its content and attributes are stored in file records in the MFT and the attributes can be resident or non-resident. These attributes contain the file’s normal data streams. It is possible for a criminal to define their own non-resident attributes and hide data in the attribute’s content, these attributes are called Alternate Data Streams (ADS) and won’t be accessed by the OS. An ADS can cause a number of problems [Mar02]:

- Virus attacks: hiding malicious executables

- Denial of Service (Dos) Attacks: creating so many ADS that the system runs out of resources and crashes
- Data Hiding: placing evidence in an ADS so it won't be found by conventional file searches.

It is therefore important for computer forensics tools to be able to locate and analyse ADS.

The \$BADCLUS file record keeps track of clusters which have been reported as bad and if the cluster address is stored in this attribute then it is identified as bad. Bad clusters won't be allocated to a file and therefore data contained in them will never be overwritten by the file system. A criminal could take advantage of this and manually alter the \$BadClus file record to set a number of normal clusters to be bad by adding them to this attribute, and then place data they want to hide in those clusters [Car05]. However virtually all bad sectors are usually found by the hard drive [SJ00] making it unlikely that the file system would need to mark clusters as bad. This makes it easier for an investigator who can process the MFT to find all clusters marked as bad and then search these clusters for potential evidence.

NTFS was designed to be flexible: the MFT is small when the file system is created and there is space after the MFT reserved for its expansion (as opposed to FAT systems where the File Allocation Table is of a fixed size) [Car02]. However there will still usually be unused space in the MFT which can be used to store hidden data although it is possible that this data could be overwritten if the file record becomes allocated [Car05]. There would be much more room to hide data in a FAT file system as the file allocation table will be large even if there aren't many files being stored.

Almost all file systems contain slack space. If a file to be stored is smaller than the cluster it is being stored in (or the block it is being stored in for Unix/Linux file systems) then the remaining space in the cluster/block is wasted. This wasted space is called slack space. It is possible for a criminal to hide a reasonably large amount of data, depending on the cluster/block size, in this slack space, e.g if the file system had a cluster/block size of 16KB and a file stored in a cluster was 2KB then someone could reliably hide up to 14KB of data. Data hidden in this way would be invisible to the file system and would be invulnerable to being overwritten by normal system usage (unless the file at the beginning of the cluster was deleted and then overwritten by a larger file) [Chu].

2.4.5 Encrypted Data

Rather than hide data, a criminal may decide to encrypt it so that its content can't be viewed. An investigator who is mainly looking for hidden and deleted files may overlook it. Encrypted files are more likely to contain evidence than non-encrypted files so their presence may help narrow the focus of an investigation, the disadvantage is that it can take an inordinate amount of time to decrypt the files depending on the strength of the encryption [Cas02].

Minor criminals may use simple (weak) encryption to conceal evidence of their crimes. One method is to use an XOR function which essentially reverses every bit, this method was used by early versions of Microsoft Word and Excel and is easy for computer forensics experts to break by simply reversing the function [Cas02].

Other criminals may use a strong encryption method that makes use of public and private keys. One simple method of breaking this encryption is to use a brute force attack which tries every possible combination of keys. This approach will work for smaller private keys, although it may take a while, but for larger keys it would take so long that by the time the data was decrypted it would be worthless. For example data encrypted using keys of 128 bits could potentially take millions of years to decrypt [Cas02].

Whilst the software component of the encryption can be strong, the human component can be much weaker. Criminals may choose a key which is plaintext and easy to remember so an investigator could simply guess it, or if the criminal chooses a key which is sufficiently complicated then they may need to write it down so searching the area surrounding the computer may yield results [Cas02].

The Regulation of Investigatory Powers Act 2000 (RIPA) allows police forces to issue a notice requiring the delivery of keys necessary to decrypt text/files. It is a criminal offence to disobey this notice unless a suspect can prove they have forgotten and/or don't have the key [SS]. Even if the key can't be found a plain text version will have probably existed on the disk at some point and it may be possible to recover that version. The file will have been stored in memory when it was being created and it may still exist in the swap file. Alternatively temporary files may have been created when the document was being written (Microsoft Word does this) and then subsequently wiped so it may be possible to recover the wiped files [Cas02]. This evidence may then be as useful as the encrypted file would have been had it been possible to decrypt it.

2.5 Layer 3: Data In Context (Files)

Once the files have been recovered, and decrypted if necessary, it is then necessary for an investigator to determine if they contain evidence relevant to the case being investigated. Often this may be as simple as viewing/reading the file to see if the contents or searching the contents of a text file for specified words or phrases. This will be most likely when the file *is* the evidence.

However there will be other times when the file provides clues about the location of other evidence, the file may be too complicated to be easily read and understood by a user or the file may contain hidden evidence which needs to be extracted and processed. In this section I will discuss how files can be processed to discover criminal actions, or events leading to criminal actions. I will also discuss how files can contain hidden data (Steganography) along with how this data can be located (Steganalysis).

2.5.1 Event Reconstruction

Criminals who download pornographic images are now using the defense that the images were accidentally downloaded by a program they weren't even aware of, a program that was hidden inside a legitimate program – a trojan horse. It is now necessary for computer forensics investigators to be able to reconstruct the events leading up to this sort of crime in order to determine guilt or innocence [CR04].

In the case of determining how a file was downloaded from the internet the first step is to analyse the internet history files in order to reconstruct past browsing activity. Unfortunately criminals may use a variety of web browsers and not all their data structures are well documented. An Internet explorer history file contains a list of activity records and each record contains the URL of the site visited, the date the user last visited the site and the length of time the user viewed that site before the browser was redirected [Jon]. This information would be useful for identifying whether the browser had been directed to a site containing illegal content. If the user had viewed an illegal site then this information could also be used to determine if the site was visited by accident in which case the user would have redirected the browser after a few seconds. Repeated visits to various illegal sites would suggest the user intentionally downloads that material unless the trojan runs every time the system is booted and downloads material from a number of sites. It may be useful to create a database of URLs associated with illegal sites so that they can be easily identified in the activity records.

Researchers have found that it is useful to construct a time line of the creation times for files with illegal content. This information can sometimes distinguish file created by a user and files created by a malicious program – files could be created much more rapidly by a program than by a user. Files with creation times that are less than a few seconds apart were probably created by a trojan horse [CR04]. Although it should be noted that a criminal could manually change these times or use a scheduling program to download files.

Digital event reconstruction is becoming important for all areas of computer forensics, not just internet activity. There are no current tools which attempt to perform event reconstruction [CS04]. One general model proposed is to use the same steps involved in physical event reconstruction; this involves identifying the individual evidence objects, producing a list of events which may have caused/produced those objects and then identifying sequences those events could have occurred

in. Hypotheses are then produced and tested based on these events [CS04]. However this is a very abstract model and no expert has yet developed an automated method for reconstructing digital events.

2.5.2 Hidden Data: Steganography

“Steganography is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message” [Wikc].

It is possible to hide data inside legitimate files in such a way that it is difficult to detect that the file contains hidden data – this is steganography and is starting to be used more by criminals to hide evidence of their crimes. Simply encrypting data in files has the disadvantage that an investigator is aware of the file and will therefore try to decrypt its contents, an investigator viewing a file that uses steganography could easily dismiss its contents as irrelevant to his/her investigation. Data can be hidden in a file of any format but the most common files used are images.

There are 3 main ways of using steganography with images; least-significant bit (LSB) embedding, transform techniques and perceptual masking methods [LD]. I will briefly cover the first 2 methods in this section as the third method isn’t covered well in the literature and due to space limitations.

Image files (except JPEG) store the colour information of each pixel in a number of bytes, this colour information is divided into the intensity of the red, green and blue colour components. The number of bytes used depends on the file format, 24-bit images store the colour of each pixel in 24 bits (1 byte for each of the red, green and blue intensities) [JJ98]. It is difficult for the human eye to detect very slight variations in colour shades, therefore a message could be encoded by the slight changes in colour in an image and it would be difficult to detect. This is essentially how least significant bit encoding works, the least significant bits of each pixel are altered to contain desired binary data. Three bits can be stored in each pixel (by using the least significant bit for each colour value) although it is possible to use the 2 least significant bits and the human eye would still be unable to discern it [JJ98]. Figure 10 is an example of 2 images, the one on the left is the original and the one on the right contains a message hidden using LSB encoding.



Figure 10: Data Hidden Using LSB Encoding [LD]

It is recommended that LSB encoding is only used with lossless image formats (.GIF, .BMP) as the message could easily be lost during the compression process used by lossy formats (.JPEG) [JJ98]. It is possible to hide data in JPEG images by using transform embedding techniques. These techniques involve modulating coefficients in a transform domain, for example with jpeg images it is possible to modulate coefficients of the discrete-cosine transform (used in JPEG compression) based on the bits of the message to be hidden and the round-off error during quantization [LD]. This method is much more complicated and can be difficult to understand but it provides advantage in terms of robustness as the message is less likely to be destroyed by manipulating the image.

Given criminals' improved ability to hide data an investigator can no simply view a file to determine whether or not it is evidence relating to their case. Therefore more scientific, and preferably automated, techniques are needed. Analysis of data hidden using steganography is called steganalysis.

The assumption that a message hidden using least-significant bit embedding won't leave any detectable signature is only valid if the number of unique colours in the image is comparable to the number of pixels but this is rarely the case. One, possibly reliable method of detecting images files with messages hidden using least-significant bit encoding is the use of close colour pairs. Two colours are close if the modulus of the difference between their red, green AND blue values is 1, two colours are unique if the modulus of the difference between their red, green OR blue values is 1. The important value identified is the ratio between the number of close colour pairs and the number of unique colours, researchers have found that this ratio is higher in images that don't contain a hidden message compared to images that do contain a hidden message. More importantly it has been found that if an image containing a hidden message is embedded with another message then the ratio doesn't change significantly whereas an image containing no message that is embedded with an image causes the ratio to decrease significantly. Images can be copied and artificially embedded, the rate of change in the ratios can then be measured and if its value lies within a certain threshold it can be assumed that the original image contained a hidden message. However a problem lies in automatically choosing a threshold value [MRMS].

Alternatively hypothesis testing methods can be used which don't require the user to choose a threshold value. One such method statistically models the hypothesis that the file contains hidden data and the hypothesis that the file doesn't contain hidden data. The hypothesis that no data is hidden is tested by using the log likelihood ratio test [SDM⁺]. Whilst this method does use a threshold it still provides good performance if the threshold is set to a default value of 0 [SDM⁺]. The method also has the advantage of not being limited to detecting least-significant

bit steganography techniques, it can detect any steganography method with a good statistical description [SDM⁺]. There are many other steganalysis techniques for use with images but they will not be discussed here due to space constraints.

Data can be hidden in audio files although doing so has the disadvantage that the human audio system is much more sensitive than the human visual system and can detect alterations in the signal more easily. One method of audio steganography is low bit encoding which replaces the least significant bit at each sampling point with message data, the disadvantages of this method is that the message can be easily destroyed by resampling, channel noise etc. [BGML96] and is therefore less likely to be used by criminals. One of the most effective methods, in terms of perceived signal to noise ration, is phase encoding which adjusts the phase of a segment of the signal with a reference phrase that represents the data, the phase of subsequent phases is adjusted to preserve the relative phases between signals [BGML96]. There are many more audio hiding techniques which won't be discussed here due to space limitations.

There has been little published on steganalysis of audio files although the hypothesis testing method used for images should also work with most audio encoding methods. Otherwise scanning high inaudible frequencies for odd distortions or patterns might reveal the existence of a message [Kre04], although it isn't clear what is meant by 'odd' in relation to distortions or patterns. Also differences in pitch, echo or background noise may raise suspicion [Kre04] but again very little has been published on steganalysis of audio files.

2.6 Current Computer Forensics Tools

There are many computer forensics tools on the market and in this section I will briefly describe the features of some of the different packages and highlight their limitations.

2.6.1 EnCase by Guidance Software

From my review of the literature EnCase appears to be the most commonly quoted and most commonly used forensic toolkit. The following analysis is based on the product description provided by the company [Sof06]. This toolkit has support for virtually every file system and OS available including DVD and CD formats. Disk Imaging features are offered along with the necessary validation to prove the copy is identical to the original and the amount of data lost due to read errors (if any) is reduced although any sector in which an error occurred will be lost. Note that when proving a copy is identical to the original it is accepted practice to ignore sectors that couldn't be read providing they aren't used during any analysis. Depending on the case being investigated it isn't always necessary to create an image of the entire device so EnCase provides the user options for imaging individual files.

EnCase allows a user to search through large amounts of data easily by providing a large number of filters which can select files as specific as encrypted files of a certain type. Advanced queries are also allowed so that the user will have a greater chance of finding evidence which is more relevant by using regular expressions or even keywords in different languages.

Whilst EnCase can only recover data to the same level of other tools it does simplify the results that are presented to the user. These results take the form of a windows explorer style interface which are easier to search through than directory listings although a computer forensics investigator would most likely be skilled enough to use directory listings easily. Guidance Software claim the tool can acquire and analyse encrypted volumes but there is no mention of how the tool deals with encrypted files.

Event reconstruction is offered although it only involves analysing time stamps and recording which events occurred in which order and doesn't attempt to offer explanations as to why specific events occurred. For internet activity analysis the pages which were viewed can be reconstructed which may be helpful for some investigations.

This toolkit doesn't provide any steganalysis features which is a major disadvantage if the only evidence relevant to a case has been hidden inside another file.

2.6.2 Forensic Toolkit by Access Data

The Forensic Toolkit by Access Data provides support for all Windows and some Linux file systems although there aren't as many as supported by EnCase. However disk imaging could be made more efficient by this tool's ability to create multiple images simultaneously and more file systems are supported by the imaging function than by the rest of the toolkit [Datc].

This toolkit also provides some advanced searching features such as being able to search for specific binary patterns, and to create user-defined filters. However there aren't as many built-in features as provided by EnCase and there is no foreign language support, nor any ability to create time lines for event reconstruction [Datb].

There is a separate tool available from Access Data which is apparently able to recover all files protected by passwords and/or encryption. This appears to be more than is obviously available from EnCase and an alternative tool uses a network of machines to decrypt files. However a disadvantage would be that this tool can cost over \$39000 which may be beyond the scope of smaller law enforcement agencies [Data].

As with EnCase there is no mention of any steganalysis features so once again important evidence could easily be missed if it is hidden inside another file.

2.6.3 ProDiscover Forensics

ProDiscover's disk imaging has the useful feature of being able to analyse and image HPAs, something which is lacking from both EnCase and Forensic Toolkit. The imaging tool has also met the standards set out in the NIST imaging tool specification which means that data recovered in this was will be almost certainly admissible in court. As with all the toolkits investigated there is no capability of detecting a DCO.

The toolkit supports the most common Windows and Unix/Linux file systems which is more than is supported by Forensic Toolkit although EnCase does support more. However because of the format used by the toolkit other tools can be used to analyse systems not supported by ProDiscover. There is no mention of any advanced searching techniques and no functionality to deal with encrypted files.

The deleted file recovery will be as effective as the other toolkits as they all appear to search the same locations for deleted data. Once again there is no support for steganalysis so evidence hidden inside other files could easily be missed [For].

2.6.4 SMART by ASR Data

SMART, like ProDiscover, allows efficient disk imaging by being able to create multiple images simultaneously although there is no mention of SMART being able to detect a an HPA/DCO. Authentication and verification of images is performed automatically [Datd] [?] [?].

SMART is apparently the only tool capable of safely mounting images of jouralling file systems although if the raw image is being analysed then it may not need to be mounted. It is unclear which file systems are supported by this toolkit but there are advanced searching options, like the use of regular expressions, for those that are. This tool provides many options once a file, which may have been deleted, is located, such as recovery and extraction from the image [?] [?] [?].

However the major drawbacks for this tool are that it only allows the file system structure to be viewed by use of a hex editor which can make the data difficult to process. Also there are no steganalysis features making some hidden evidence unrecoverable.

3 Requirements and Analysis

The main aim of this project is to develop a suite of computer forensics tools which are jointly capable of analysing a variety of hardware storage devices and recovering deleted/hidden data. The reasons why each tool is necessary will be given in the corresponding section. In this section I also describe the requirements that the tools need to satisfy in order for the project to be successful.

3.1 The Tools

The tools I'll develop will perform the following functions:

- Analysing a hard disk and detecting HPAs, DCOs and bad sectors with a reasonable degree of accuracy,
- Creating a bit-stream image of a hard disk and verifying the copy
- Mapping the systems logical partitions and locating hidden data with a reasonable degree of accuracy,
- Locating files contained in a file system, recovering deleted files where possible, and reconstructing fragmented files,
- Displaying the contents of an encrypted file (where possible) and at least identify that the file is encrypted,
- Reconstructing computer events which occurred before a crime was committed,
- Detecting the use of steganographic methods, with reasonable accuracy, and extracting the data hidden using those methods,

3.1.1 Analysing a Hard Disk and Creating an Image

The first step in any computer forensics investigation is to ensure that the original data cannot be modified during the analysis. This is achieved by making a bit by bit copy, i.e. an image, of the original disk which is stored on a separate system. In section 2 I highlighted the advantages of storing the image in a file rather than simply writing the bits directly to another disk and for these reasons the tool will create the image in a file.

However the accepted practice of creating an image involves reading a number of sectors from the disk and writing them to the target image file until the entire disk has been copied. In order to do this it is necessary to have access to two pieces of information; the number of bytes in a sector - which is usually 512 but can vary between disks, and the size of the disk. The number of bytes per sector needs to be known because there are no in-built functions in the C/C++ or Java programming languages which can read a sector directly from the disk therefore the required number of bytes will need to be read instead. Also, as will be seen later, many data structure used by partitioning and file systems reference another part of the system by their sector address and as the image will simply consist of a number of consecutive bytes the number of bytes per sector will be used to calculate how many bytes need to be skipped when accessing the desired data.

The size of the disk doesn't necessarily need to be known as one feasible approach to imaging would be to simply read bytes from the target disk until the end of the disk is reached. The next read operation would then cause an exception that the tool could handle without causing a fault. However this could be considered bad programming practice and therefore the size of the disk will be used along with the number of bytes per sector and the number of sectors per read (supplied by the user) to determine how many read operations are needed to copy the entire disk and only that number of operations will be performed. This means that one requirement of the system will be that the tool must be capable of gathering basic disk information such as its size.

I have already mentioned the various locations on disk in which data may be hidden. One way of analysing the raw binary data would be for an investigator to manually read the bits, by using a hex editor, and determine their meaning using a list of data structures. However this could easily take one person a long time (1 year per GB if 32 bytes are examined per second) and it would be a very tedious job making it easy for the investigator to make mistakes. Rather than just displaying the data in hexadecimal or binary form the tool will need to store a list of data structures known to exist for different types of disk and then process them in the same way a human investigator would.

This process is fine so long as known disk type, such as ATA, is being investigated as the methods and data structures will be known. If an unknown disk type is encountered then the programmed analysis methods won't work. It is therefore important for the tool to be expandable, by that I mean it should be possible to provide the system with a file containing data structures and methods for investigating a different disk type which can then be used by the system during analysis.

Not every disk supports the use of HPAs or DCOs. Older disks don't provide these functions or even the commands that would be used to find the hidden areas and if the tool tried to detect a HPA/DCO using the normal commands then an error would be produced. It will be therefore necessary for the tool to detect the capabilities of the disk and adjust its analysis strategy accordingly (e.g. don't try to detect a HPA/DCO on an older disk). However it may not be possible for the system to detect the disks capabilities in which case it should be possible for the user to input the capabilities manually, e.g. by selecting the appropriate option from a combo box.

Another problem with older disks is that they used a different addressing scheme to locate sectors. This method was CHS addressing (the details of which aren't important) and is no longer used as it limited the size of hard disks to 8.4GB, the addressing scheme used today is logical block addressing (LBA) [SJ00]. If an older disk is being investigated then the tool would need to be able to translate between the two addressing schemes. The BIOS of the system being used in the investigation may also not support CHS addressing in which case there are two options; access the disk directly instead of via the BIOS or use a system with an older BIOS. Whichever method is chosen the important point is that the tool being used should support older addressing schemes.

The disk manufacturer performs detailed inspections of the hard disk to find any bad sectors caused by mechanical or magnetic faults. Ideally the disk analysis tool should be able to oversee the same checks to find sectors which are definitely bad and which have been marked as bad by the system/user. However these sorts of checks require specialist tools that I won't have access to so this requirement will be considered beyond the scope of the project.

During the imaging process the tool may be unable to read certain sectors, in which case it should write 0s to represent those sectors and make a note of the location in a log. This option has been chosen because it is the standard practice of professional examiners and is acceptable in court providing the data in the image corresponding to those sectors which couldn't be read aren't used during any analysis. The image should be written to a file chosen by the user, the disadvantages of writing to another disk were discussed in the literature review. The user should also be able to choose the number of bits that are copied at a time (e.g. the user could choose to copy 10 sectors, or 5120 bits, at a time) although an upper bound should be placed on this number. This is because a large amount of data could be lost if there was a read error when this number was high. For example if 1000 sectors are being copied at a time and there was an error reading 1 sector then 999 sectors may be regarded as bad also and written with 0s. An alternative would be for the system to be able to identify the individual sectors that couldn't be read and only write those as 0s, this is the preferred solution.

It will not always be a single disk that is being analysed, sometimes investigator's will encounter RAID (Redundant Array of Inexpensive Disks) systems. The tools should be able to process these disks in the appropriate manner although it could be difficult for the tool to identify which data was stored on which disk, i.e. which disks were used as backups, which order the disks were written to etc. However because high level RAID system are only found in servers it would be difficult to test the tools ability to cope with such systems. As a result I will assume that dealing with RAID is beyond the scope of the system.

For the purposes of developing an example toolkit for this project it will be assumed that a dead acquisition is being carried out which means that the suspect system was powered down when it was encountered so the hard disk could simply be transferred to the users machine and copied. However it may often be the case that the target system is still active in which case shutting the computer down in order to use a dead acquisition process may result in evidence being lost such as the transient data in memory. In these situations a method of memory acquisition would be required to retrieve data from memory and a live acquisition process, such as connecting the suspect system to the users computer would have to be used. However many methods of memory

acquisition require special hardware [CG03] which I won't have access to and live acquisition methods add complexity in having to manage a network between the suspect and user system. For these reasons live acquisition methods won't be implemented during this project.

Figure 11 shows the steps involved in analysing a disk and creating the disk image:

Figure 11: Task Hierarchy for the Disk Imaging Tool

3.1.2 Partition Analysis

Once an image has been acquired it is then necessary to find and recover deleted (and undeleted) files. However these files will be contained inside a file system so it is obviously necessary to find the file system before this can happen. File systems, on most systems, are contained inside partitions of which there may be several so the first step is to be able to analyse the partition information. This is why the next tool to be developed will be a partition analysis tool.

The first, and most obvious, requirement of the partition analysis tool is that it must be able to map the partitions and present the information to the user. However the partition table (for many systems) is contained inside the first sector of the disk, therefore it will be necessary for the tool to know how many bytes are in a sector so that it knows how many bytes to read. This information will have been gathered and stored by the disk imaging tool so the first step of analysing a partition will be to parse the disk information file to find the information required for the analysis. It will also be necessary to find the size of the disk from this file for use when finding slack space.

During a computer forensics investigation every procedure and result must be well documented. The tool should therefore also be able to store the partition information in a log file along with the date and time the analysis was carried out. When presenting the information to the user it might be helpful to display the information in a diagram to make the data easier to understand although this isn't a critical requirement.

The tool must be able to detect volume and partition slack, record the location of this slack space and record this information in the appropriate log file along with the reason for identifying it as slack space. This reason would consist of the total size of the partitions in the volume compared to the total size of the volume.

There may be occasions when one or more of the partition tables are corrupted and can't be used. The tool should be able to identify that some of the tables are unusable, make a note of which tables are missing, and then recover the partition information using other methods such as searching for known signatures. In the literature review I mentioned that searching for signatures often yields many false results. The tool should make a note of all the results in the appropriate log and then filter out the results which are definitely false, again noting these results in the log along with reasons why they're false. The remaining results which may be positive should be recorded in a log.

There are many different partitioning systems and MS-DOS is just one example, modern 64-bit systems and some Unix/Linux systems use completely different methods. The system being developed is simply an example system so only be initially able to analyse DOS partition systems is acceptable.

Figure 12 illustrates the tasks involved in analysing the partition system.

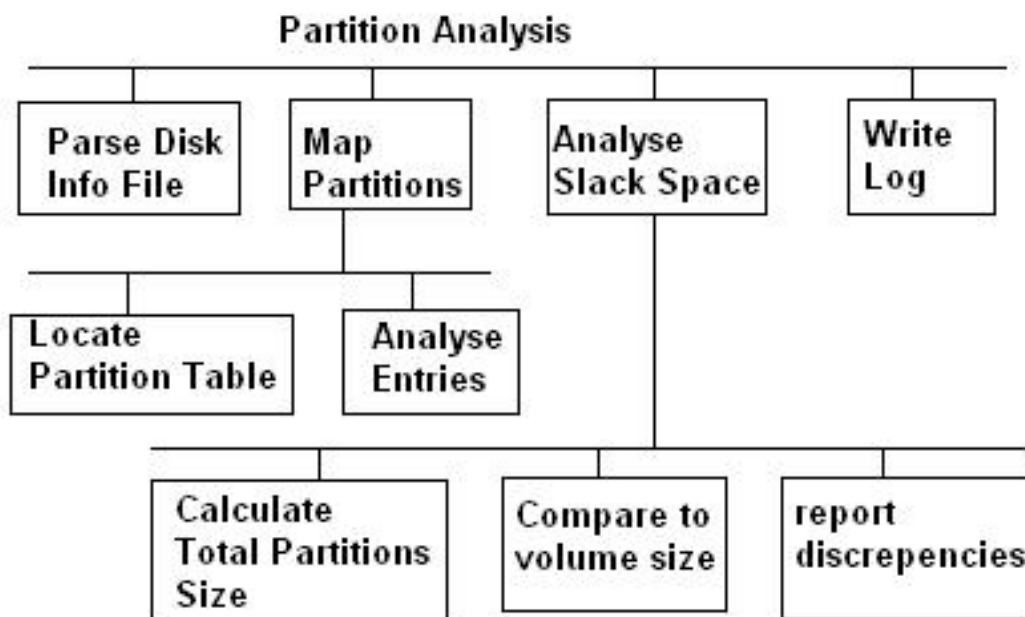


Figure 12: Task Hierarchy for the Partition Analysis Tool

3.1.3 File System Analysis

Now that the file systems have been found the tool should now be able to analyse them and recover files. There may be many file systems and the user may only want to analyse data from one of these file systems. Therefore the first requirement of the tool will be to parse the partition information file created by the partition analysis tool and present the user with a choice of file systems to analyse.

The initial requirement for the file system tool is to be able to record the type of file system (i.e. FAT, NTFS etc.) and the directory structure it contains for all current non-deleted files. This information should both be stored in the log file and presented to the user. Again it could be helpful if the information was presented to the user in the form of a diagram but this isn't a critical requirement as the user should be able to quite easily understand directory listings. The tool must also record basic file system information such as the number of sectors per cluster, the size of the file system and other file system specific information.

The tool must record which clusters are marked as good and which are marked as bad. The bad clusters should be analysed for possible evidence although this will depend on the kind of information the user is searching for. In other words the tool should include bad sectors in any searches.

One problem is that many file systems, such as FAT and NTFS, store files in clusters rather than sectors and it will be necessary to identify these clusters before the files can be recovered. File system generally have data structures which contain this information along with other general file system information such as the date it was created that might be useful to an investigation. It is therefore necessary for the tool to be capable of analysing the appropriate data structure(s) and presenting the information to a user.

Now that the basic file system information is known files containing evidence can be recovered. Whilst there has been some work on creating models for automated analysis of file systems some user interaction is still required. The user must be able to search the directory structure for files

with names containing specified keywords or files with contents matching specified keywords. The tool must also provide the option of searching for specific file types which should be located by using both the extension in the file name and the file's signature, although if the values don't match then the signature should be trusted rather than the extension. The details of the search should be noted in the log (i.e. the keywords, the file types searched for, the date and time of the search etc.) along with the results which will contain the file name and directory address. It is necessary to be able to search for files because a typical system could contain hundreds, or thousands, of files which may need to be analysed in detail to determine if they can be considered as evidence. If an investigator could fully examine a text/pdf file in 30 minutes and the system contained 1000 files, many of which are irrelevant for the investigation, then it would take over 20 days to be able to gather the required evidence. Alternatively searching for files containing certain keywords is more likely to find the relevant files more quickly than manually searching.

The tool should be able to find and recover a reasonable amount of hidden data from the file system. It will do this by first identifying possible locations in which data could be hidden and then recording these locations in the log along with reasons why they were identified. One example of this might be when the number of sectors in the file system isn't divisible by the number of sectors per cluster. These locations will also be used during the searches conducted by the user.

As the data being analysed is simply a binary file containing the disk image it will be difficult for the user to view a file's contents by just using this image file. The tool must therefore be capable of extracting the bits from the image which correspond to the file the user wants to view and place them in a file of an appropriate type, e.g. extract the bits of a jpeg image and place them into an empty jpeg file. I have managed to create a program which can place extracted bits into an empty file although it is currently necessary for the user to input the type of the file.

The tool must be capable of recovering a reasonable amount of deleted data from the system. The term reasonable is used here as some data cannot be recovered without use of specialist equipment which I won't have access to. The details of the recovered files, including their original directory address, must be stored in the log and be presented to the user. It should be noted that some of the data may have originated from other locations such as slack space or the windows swap file in which case the directory address stored would refer to these locations. The tool must allow the user to conduct the same sort of searches as with non-deleted data. It must also identify cases where data wiping software has been used and record the signature of the tool and locations which the tool has wiped.

If fragments of files are recovered then the tool should ideally be capable of reconstructing the files using various methods. However as it isn't known whether this is possible I won't make it a critical requirement.

There are many different types of file systems (FAT, NTFS, UFS, Ext, HFS, Reiser etc.) and ideally the tool should be capable of analysing all of them using their known data structures. However this may not be possible in the limited time available and only an example system is being developed so limited capabilities are acceptable. I will make it a requirement that the tool is capable of analysing FAT systems and is also capable of being expanded to support other file systems.

Figure 13 shows the tasks involved in analysing a file system.

3.1.4 Encrypted Files

It is possible that there may be a relatively large amount of time in which to conduct a computer forensics investigation and any encrypted files encountered may only use weak encryption. Therefore the tool must offer the option of performing a brute force attack on encrypted files. The details of this process should be recorded in the log and when/if the file is decrypted the key which decrypted the file should be recorded along with the plaintext version of the file. If the user decides that a brute force attack isn't feasible then the tool must be capable of recovering deleted plaintext versions of the file if possible and recording the location in the log. The actual file would be extracted from the image.

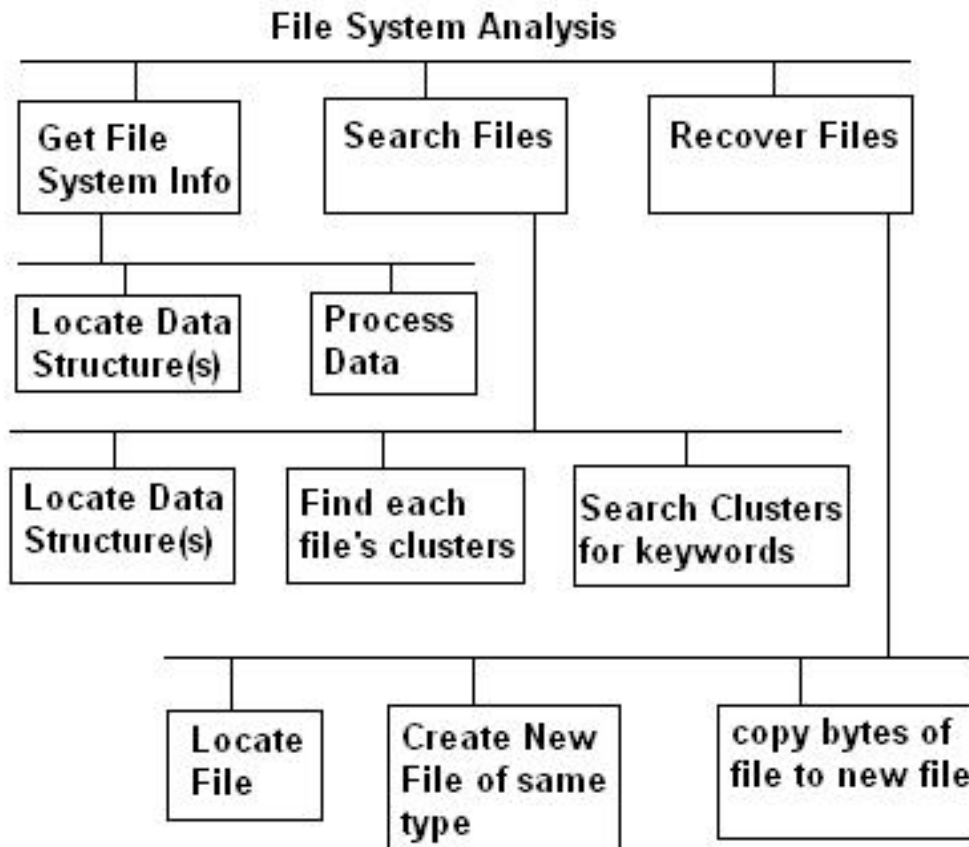


Figure 13: Task Hierarchy for the File System Analysis Tool

3.1.5 Event Reconstruction

Event reconstruction usually requires that the system and/or user files are available to be processed and can therefore only be carried out once the file system analysis tool has been developed.

The tool needs to be capable of performing basic event reconstruction using the file system's temporal information, i.e. by using the created, accessed and modified times for files/directories. The information should be organised into a time-line of possible events and then recorded in the log and presented to the user. Although it should be noted that this the temporal information used may not be accurate. If a log file is available then it should also be analysed by the tool to reinforce the time-line and any discrepancies found should be recorded in an appropriate analysis log file.

The tool also needs to be capable of reconstructing internet activity, if the necessary history files are available, and again making a time-line of the data. In this case it will be necessary to record which URLs the suspect visited and the time the browser was viewing that page. A desirable requirement for the tool is that it would be capable of automatically detecting URLs of pages which contain illegal content.

3.1.6 Steganalysis

The tool needs to be capable of detecting the use of simple steganographic techniques on images and audio files with a reasonable degree of accuracy. By simple techniques I mainly mean least-significant bit embedding as it is the most commonly used although the tool should detect other

simple methods. The tool should record which files are tested and the results of each test along with appropriate time stamps. Once files containing hidden messages have been identified then the tool should be capable of extracting the messages. These messages should be displayed to the user and recorded in the log.

It would be desirable for the tool to be capable of detecting the use of any steganographic technique but this would be over ambitious given the amount of time available and the current capabilities of investigators.

3.1.7 General Requirements

By definition the whole point of computer forensics analysis is that the evidence recovered will be used in a court of law. There are various rules and procedures which must be followed in order for evidence to be deemed admissible. I have briefly highlighted some of these procedures in the literature survey such as verifying that an image is identical to the original disk data. Following on from this a requirement of the tools developed will be that they adhere to the appropriate procedures so that the output they produce can be used.

3.1.8 Non-Functional Requirements

The toolkit developed will need to be portable as it could be necessary to use it on a variety of machines. For the most part I will therefore be using the Java programming language as the programs developed could be used on any machine which has the java virtual machine installed. However there will be a problem with the disk imaging tool as Java would be unable to perform the necessary low-level disk access. For this tool I will use the C++ programming language as it overcomes these difficulties but I would need to make sure this tool still has the required portability.

The system developed would obviously need to be secure although this can be achieved by using a stand alone machine which isn't connected to any outside sources such as the internet. However where this isn't possible I will aim to make the tools as secure as I can.

It is very difficult to develop non-functional requirements related to the time taken to complete the analysis as every case could be different in terms of the amount of analysis necessary and the amount of time available to conduct the investigation. These requirements may be added at a later date. The system will need to be reliable as any analysis interrupted by system failure will need to be repeated and some of the analysis could take a long while, e.g. disk imaging may take a few hours depending on the size of the disk.

3.2 Specific Requirements

Requirement No	Name	Description	Type
1	Analyse Disks	See Disk Imaging Tool Requirements	Essential
2	Analyse Partitions	See Partition Analysis Tool Requirements	Essential
3	Analyse File Systems	See File System Analysis Tool Requirements	Essential
4	Analyse Files	See File Analysis Tool Requirements	Essential

Figure 14: Table 3.1 General Project Requirements

The requirements of the Disk Analysis Tool are shown in figure 15:

Requirement No	Name	Description	Type
1	Get Disk Info	The System must be able to gather information about the target disk, such as the number of bytes per sector, the size of the disk etc. This information should be displayed to the user on request and also be stored in a specified file.	Essential
2	Create Disk Image	The System must be able to create an image of a target disk that is saved in a file specified by the user. The image must be created by copying n sectors at a time, where n is specified by the user	Essential
3	Create Imaging Log	The System should create a log of the imaging process detailing how many bytes were copied, from which disk they were copied, the time they were copied and the file they were written to. Any read errors should also be recorded along with the LBA address of the sector(s) that couldn't be read and the time of the read error.	Desirable
4	Verify Image	The system should be able to verify that the image created is identical to the original disk. This should be done by calculating a message digest of the image and the disk and comparing the values to make sure they're identical.	Desirable

Figure 15: Table 3.2 Disk Analysis Tool Requirements

The requirements for the partition analysis tool are shown in figure 16:

Requirement No	Name	Description	Type
1	Get Partition Info	The system must be able to gather the partition information from the specified disk image. This information must include the type and address of each partition the image contained. The gathered data should be displayed to the user if requested and also saved in a specified file.	Essential
2	Create Analysis Log	The system should be able to create a log of the analysis highlighting why particular results were obtained.	Optional

Figure 16: Table 3.3 Partition Analysis Tool Requirements

The requirement of producing an analysis log as described in figure 16 has been designated as optional because many courts have ruled that a forensic tool cannot be considered as an expert. This means that a computer forensics expert would need to be able to explain why the tool obtained particular results without relying on the tool thus making the requirement slightly redundant. However it would be necessary to make a note of what analysis was performed on the disk (or image) and when but it isn't necessary for the tool to do this as the examiner could manually create an analysis log.

Figure 17 contains the requirements for the file system analysis tool.

The requirement of creating an analysis log is optional for the same reasons as the requirement

Requirement No	Name	Description	Type
1	Get File System Info	The system must be able to gather information about a specified file system. The file system will be chosen by the user selecting a partition after loading the partition information file already created using the previous tool. The retrieved information should be displayed to the user if requested and also saved to a specified file.	Essential
2	Search Files	The system must allow the user to search the specified file system for files whose contents contain given keywords or whose name matches the string entered. Another option must be to list all files. It should be possible to filter all searches to only include files of certain types or all deleted files etc. Results of the search must be displayed on the screen and it should also be possible to save the search criteria and results in a given file.	Essential
Requirement No	Name	Description	Type
3	Recover Files	It must be possible for a user to recover files from the image into another file on another volume (possibly the same volume). The system must allow the user to select a file by entering its name.	Essential
4	Event Reconstruction	The system should be capable of using the accessed, modified and created times of all the files in the specified file system to create a timeline of system activity. This timeline should be easy to understand and displayed to the user if requested. The data should also be stored in a specified file.	Desirable
5	Create Analysis Log	The system should produce a log of any analysis done using the functions described above. This log will include details of the analysis performed along with the necessary timestamps and should be saved in a file specified by the user.	Optional

Figure 17: Requirements for the File System Analysis Tool

of the partition analysis tool. The requirement of using all the file's timestamps to create a time-line is desirable, rather than essential, because it is relatively simple for a criminal to manually alter these timestamps. As a result the time-line created could be unreliable and not used in many investigations, therefore it isn't an essential feature.

The steganalysis requirement described above is only desirable as it can't be guaranteed that the algorithm to be used will be able to detect the use of steganography. Whilst the authors of the algorithm did perform some testing they found that it didn't work successfully on every occasion. One part of this project will be to test the algorithm, which will be done by implementing the

Requirement No	Name	Description	Type
1	Steganalysis	The system should be able to detect, with a reasonable degree of accuracy, whether a specified file contains data hidden using least significant bit steganography methods. The system should then be able to recover the hidden information and display it to the user and save the data to a file.	Desirable
2	Internet Activity	The system should be able to interrogate a specified internet history file and determine which URLs have been accessed by the user, on what date and how long the browser was directed at each particular site. This data should be presented to the user, if requested, and saved in a specified file. The recovered list of URLs should be checked against a database of sites known to contain illegal content and the user should be informed of any matches made.	Desirable
Requirement No	Name	Description	Type
3	Encrypted Files	The system should be able to decrypt an encrypted file specified by the user on most occasions. A decrypted version of the file should be stored in a location specified by the user.	Desirable

Figure 18: Requirements for the File Analysis Tool

steganalysis requirement into the file analysis tool.

The internet history requirement is desirable due to the number of different web browsers that are available, each of which create their own unique history file. It would be infeasible (in the time available) to create a file which is capable of analysing all these different file types, although a serious attempt will be made to create a tool capable of analysing Internet Explorer history files.

Many encryption algorithms that are currently used are very sophisticated and complex and can take a team of experts years to find a weakness that can be used to break the algorithm. The only other alternative is a brute force attack which is often infeasible due to the amount of time that would be required. Because of these difficulties the requirement of being able to decrypt most files is desirable rather than essential as it may not be possible to achieve this goal.

The project will be considered to be a success if all the essential requirements and most of the desirable requirements are met.

A separate set of functions will be created to fulfil each requirement described above and these functions will be individually tested once they are completed. This testing will be a test-first category/partition method and involves creating a test set before the function is implemented. The test set will be generated by considering each input to and output from the system which are then partitioned according to a range of values considered to be acceptable, for example the number of sectors per read input to the disk imaging tools could be partitioned into values greater than 0 (which are acceptable) and values less than 0 (which aren't acceptable).

Once each set of functions have passed all the tests they will then be integrated a higher-level functional unit, such as the disk imaging tool, and then tested again using a similar method to ensure that all the components work in conjunction with each other. This testing will be done with each of the tools and when it is completed they will be integrated into the complete system and

once again tested to make sure there are compatible. This final system testing will be performed by attempting to analyse a complete disk and checking that the results are correct. A number of test images are available on the Internet, which are provided with the expected outcome from an analysis tool and will be used for the system testing. (Cite website here?).

The system will be evaluated by use of acceptance testing were a user will use the system to perform a complete analysis of multiple disks to check that the system can complete these tasks within a reasonable amount of time. The times taken by the system to carry out the various tasks will also be compared to the times taken by some of the commercially available forensic computing tools to see if the performance of the system I develop falls within an acceptable range set by commercial tools.

4 Design

Rapid prototyping will be used for the design due to the limited amount of time available. This method will enable the system to be developed incrementally, making sure that there is a working system at all times. This has the advantage of reducing the amount of system testing that is required when the final functional unit is completed. Some other methods, such as the discovery method, aren't appropriate in this scenario as there is no real client, making some of the procedures redundant. Also, the discovery method involves over 40 steps, which would take a long time to complete, and produces a large amount of documentation which I don't feel is necessary for this particular project. Only a limited amount of documentation is needed because many of the data structures used will be identical to the data structures defined in various file system specifications (e.g. the partition table used by most windows systems).

4.1 X-Machines

I will use the method of creating X-Machines from extreme programming as it is useful in use interface design by describing the various screens that the user will use to perform different tasks along with the input they expect to provide. They also gives some indication of when error messages need to be produced, thus providing some assistance when testing the system.

On the following pages are the X-Machines that have been produced for the more complex functions the system needs to perform. The remaining X-Machines can be found in appendix A.

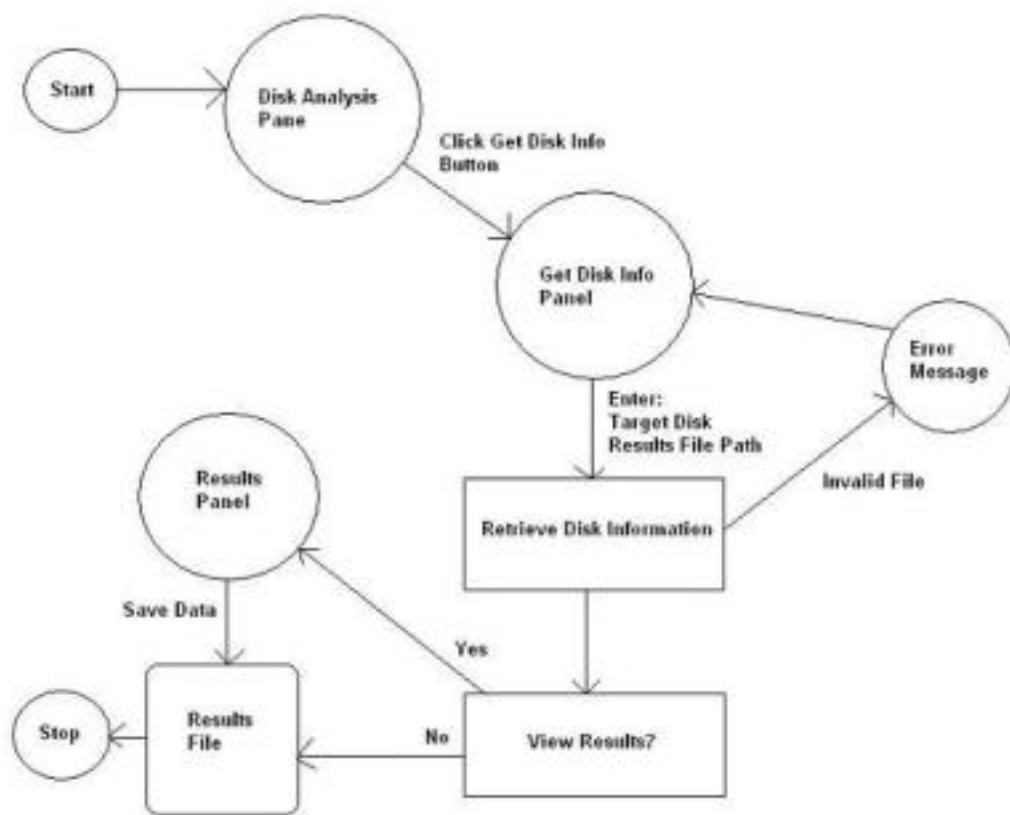


Figure 19: X-Machine for the 'Get Disk Info' function

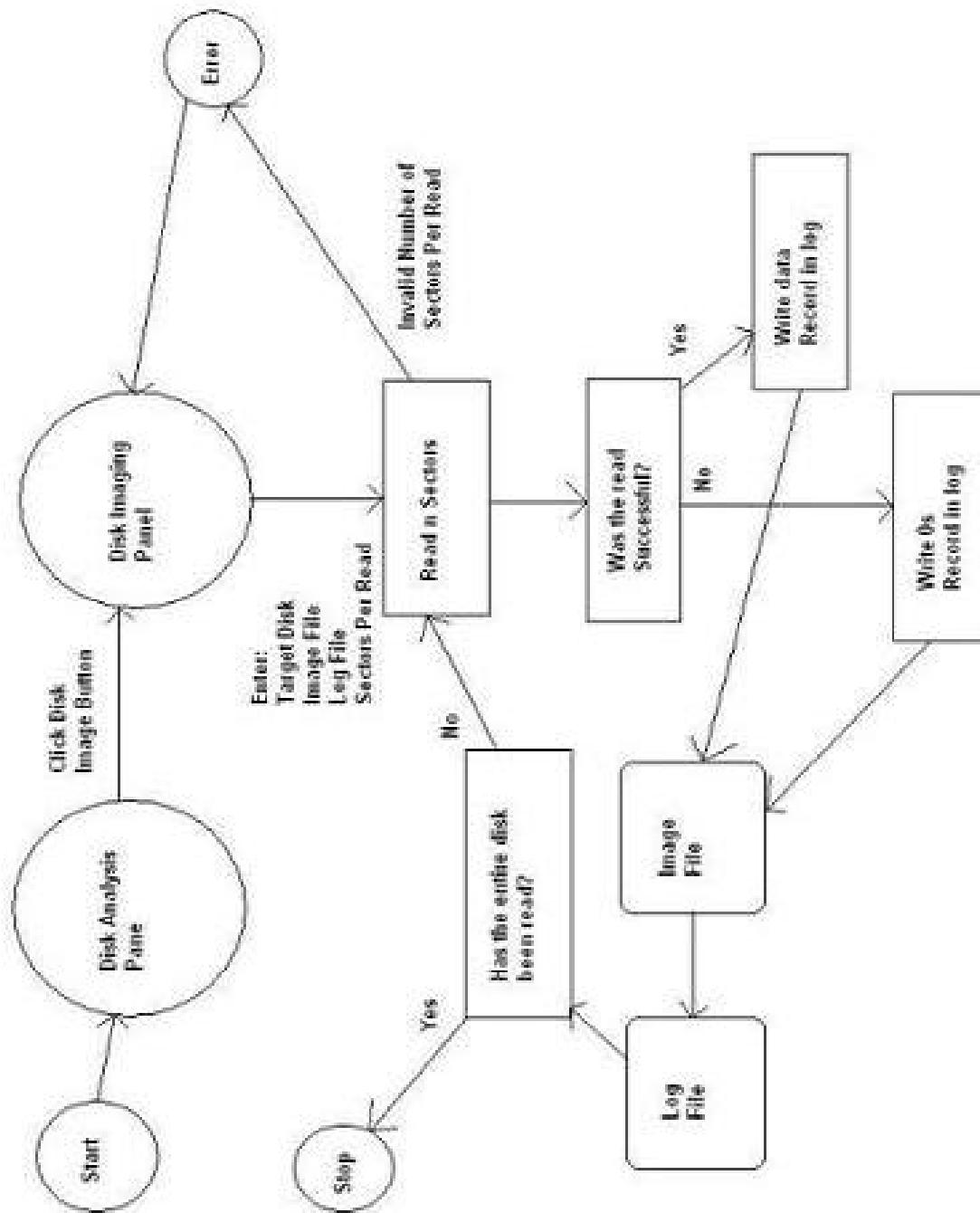


Figure 20: X-Machine for the 'Create Disk Image' function

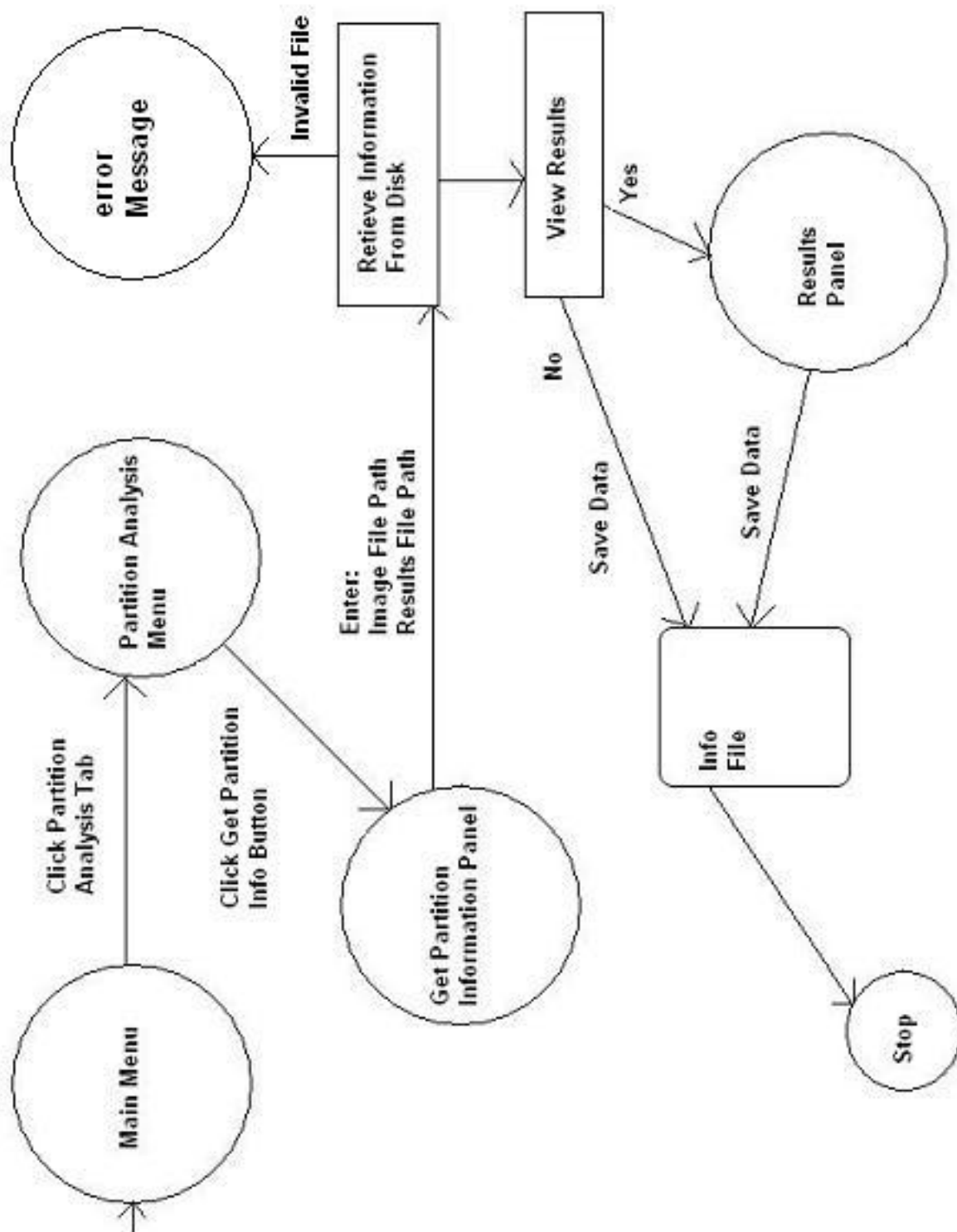


Figure 21: X-Machine for the 'Get Partition Info' function

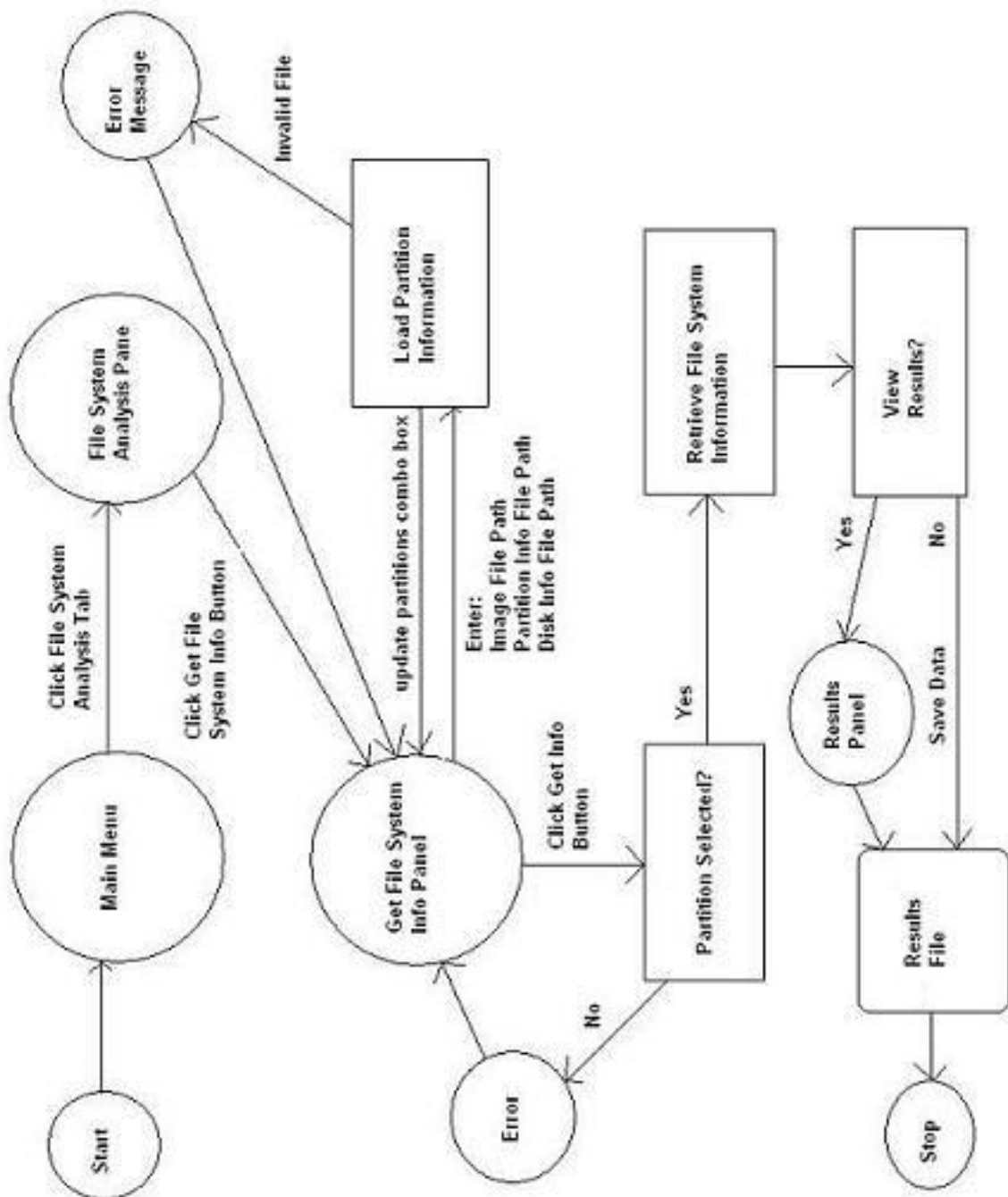


Figure 22: X-Machine for the 'Get File System Information' function

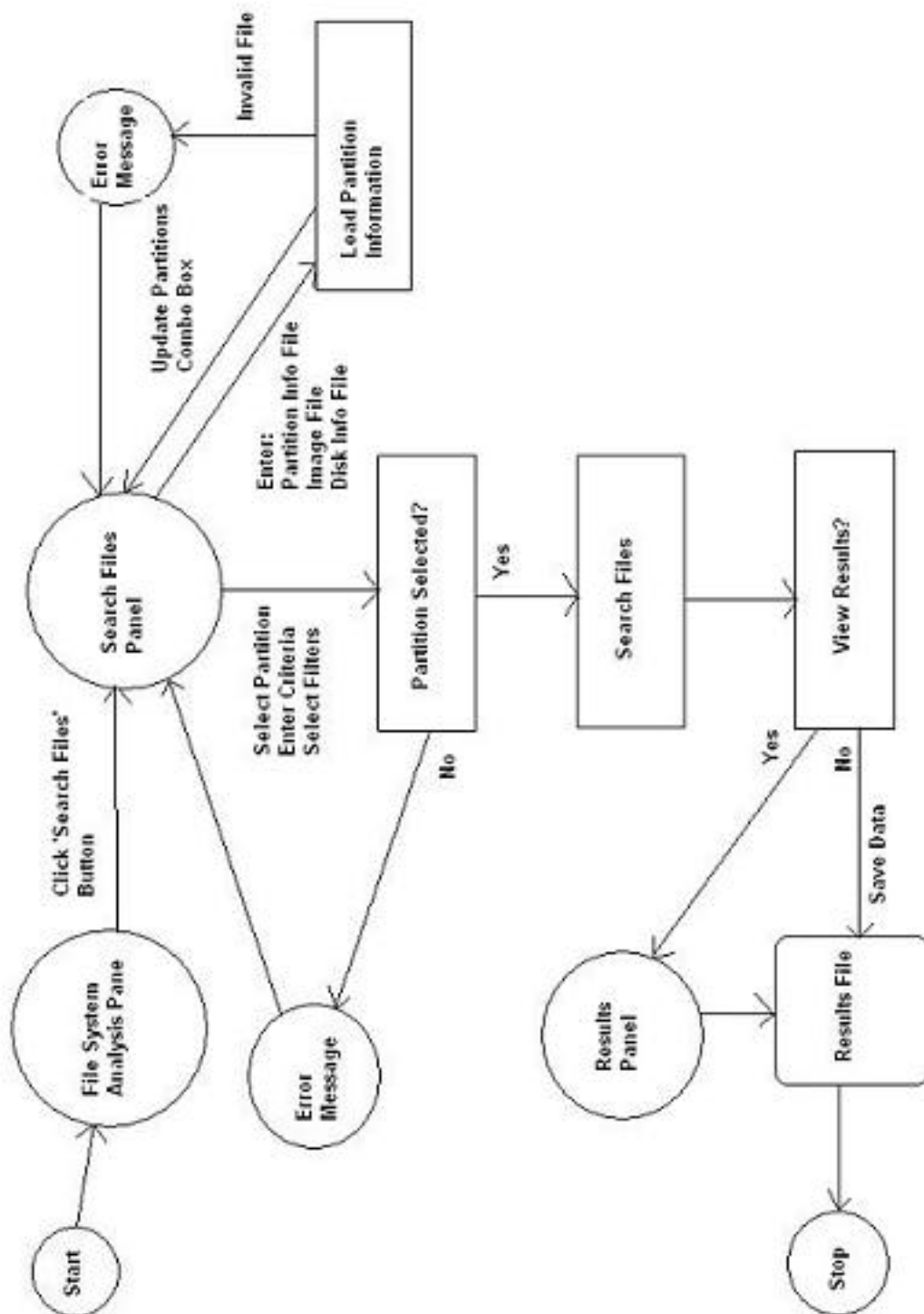


Figure 23: X-Machine for the 'Search Files' function

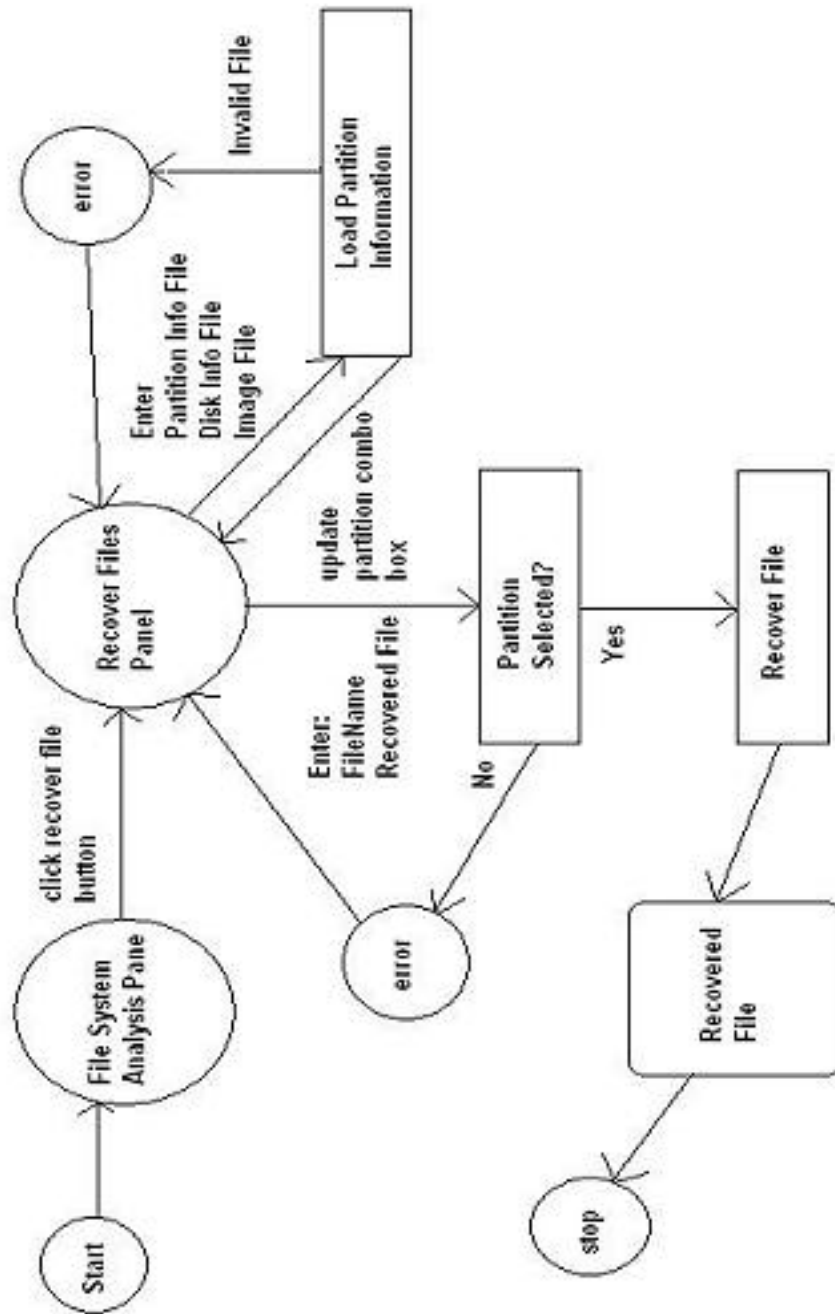


Figure 24: X-Machine for the 'Recover Files Image' function

From the X-Machines I produced the following prototype for the system:

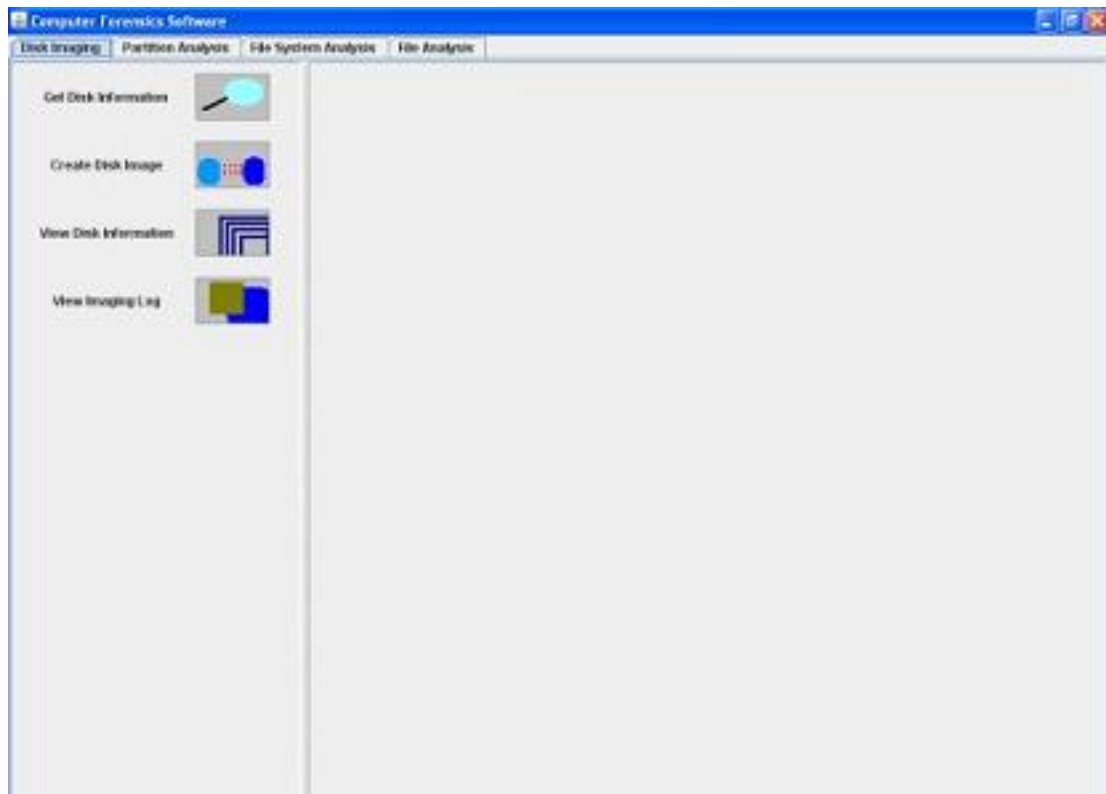
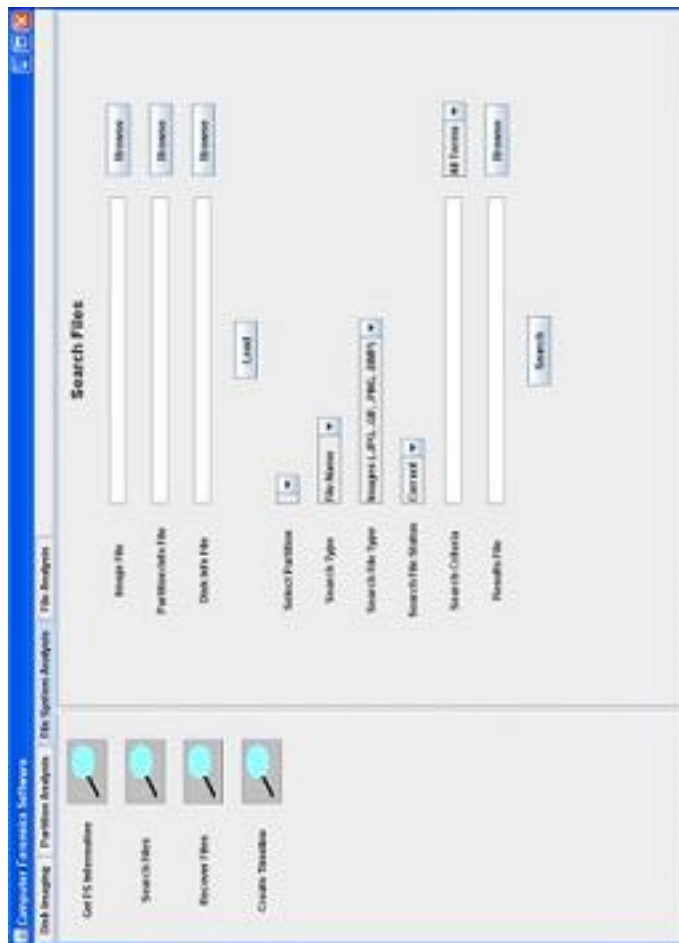


Figure 25: Prototype of the main menu

The images on the next page are a screen shot of the search files screen and the create disk image screen. Larger, more detailed, screens of the prototype are included in appendix A.



4.2 Partition Analysis Tool

The disk image acquisition tool will only need to make use of the basic data types built into the languages that will be used. However the partition analysis tool will need to use some custom built classes that represent the structures used by the partition system of the disk acquired. For analysing DOS partition systems it will be necessary to define a DOS partition table structure along with the partition entries that appear in the table. In the event that the system contained an extended partition - an extended partition table and secondary partition entry will also need to be defined. All of these data types will be used by a DOS partition analyser class which will store the details in an appropriate file. The relationship between these classes, along with the methods and variables they'll use are illustrated in figure 26.

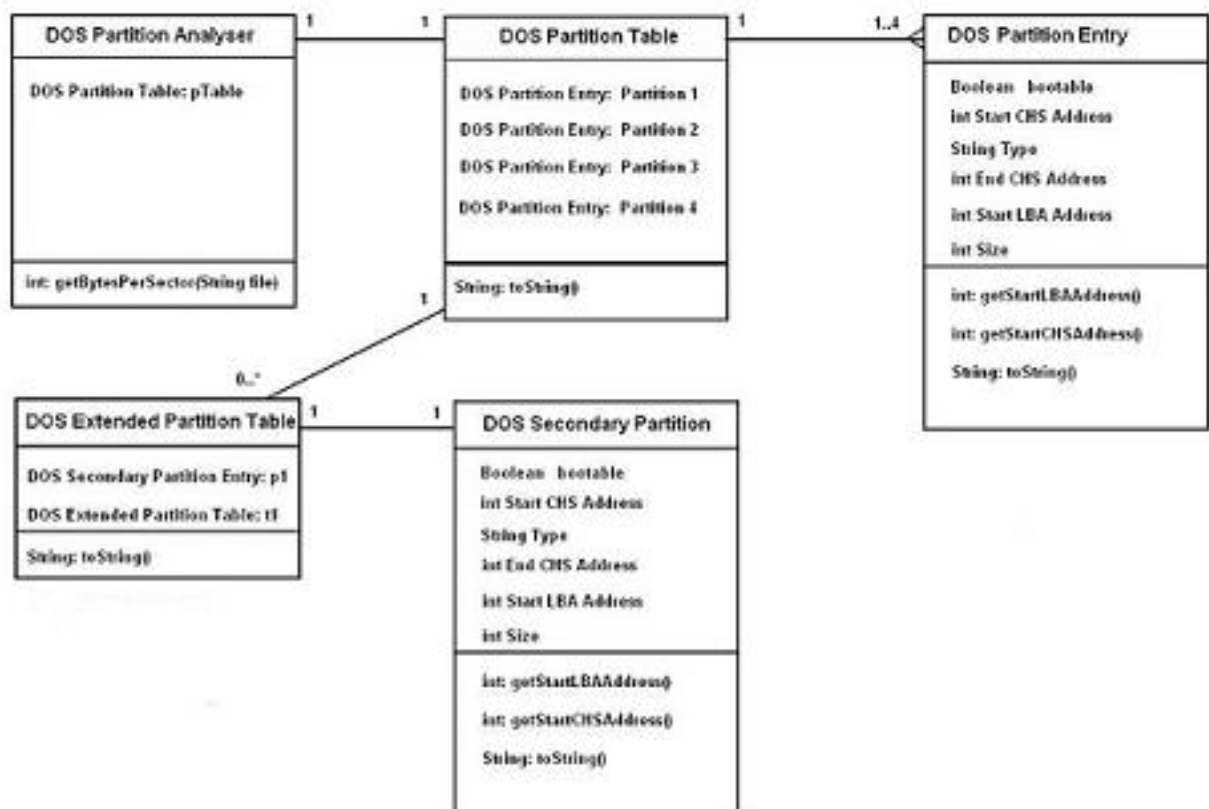


Figure 26: Relationship between classes used by the partition analysis tool

4.3 File System Analysis Tool

The file system analysis tool will primarily analyse file allocation table(FAT) file systems, although it will be possible to expand the tool to support other systems, and will need to implement some of the data structures used by the FAT system. The first data structure needed is the FAT boot sector and contains basic information about the file system which will be extracted by the tool. The boot sector varies between FAT16 and 32 which means that separate data structures will be needed, however there are many commonalities so it is possible to create a base class which the more specific classes will inherit. This relationship is shown in figure 27 and figure 28 illustrates the details of each class. Note that the designs presented have been chosen because they are virtually identical in design to the data structures used by FAT file systems (as given in [Car05]) with the only exception that file systems don't use Java data types to store the data. This isn't

essential, for example the data field containing the time a file was created in tenths of seconds probably won't be needed for analysis, but the level of consistency between the tool and the file system provided by doing this may be useful when the tool is expanded. For example later versions of the system may require the time a file was created in tenths of seconds.

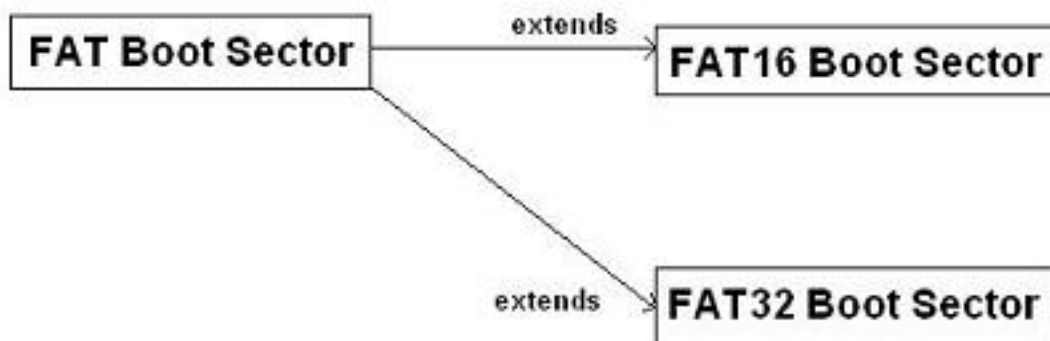


Figure 27: Relationship between the different boot sector classes

FAT Boot Sector	FAT16 Boot Sector	FAT32 Boot Sector
String: OEM Name int: bytes per sector int: sectors per cluster int: reserved area size int: number of FATs int: root dir max files int: sectors in file system String: Media Type int: FAT size int: sectors per track int: number of heads int: sectors before partition int: sectors in file system (32-bit) String getOEMName() int getBytesPerSector() int getSectorsPerCluster() int getReservedAreaSize() int getNumberOfFATs() int getRootDirMaxFiles() int getSectorsInFileSystem() String getMediaType() int getFATSize() int getSectorsPerTrack() int getNumberOfHeads() int getSectorsBeforePartition() int getSectorsInFileSystem32() String toString()	int: INT13h Drive Number int: extended boot signature int: volume serial number String: volume label String: File System Type int: signature int getINT13hDriveNumber() int getExtendedBootSignature() int getVolumeSerialNumber() String getVolumeLabel() String getFSType() int getSignature() String toString()	int: FATSize int: FATPolicy int: minorVersionNumber int: majorVersionNumber int: rootDirCluster int: FSINFOCluster int: bootSectorBackupSector int: INT13h Drive Number int: extended boot signature int: volume serial number String: volume label String: File System Type int: signature int getFATSize() int getFATPolicy() int getMinorVersionNumber() int getMajorVersionNumber() int getRootDirCluster() int getFSINFOCluster() int getBackupSector() int getINT13hDriveNumber() int getExtendedBootSignature() int getVolumeSerialNumber() String getVolumeLabel() String getFSType() int getSignature() String toString()

Figure 28: Details of the boot sector classes

The most important data structure in the FAT file system is the file allocation table itself as it is used to locate the next cluster in a file and to determine which clusters are unallocated. The FAT simply consists of a number of equal sized entries (12 bit entries for FAT12, 16 bit entries for FAT16 etc.) which are essentially just numbers corresponding to the address of the next cluster. Some numbers are reserved to represent the cluster as being unallocated (0), the end of the file (any number over fffff8h for FAT16) or damaged (ffffff7 for FAT16). One way of representing the FAT would be to simply use an array of integers in the class which is analysing the file system and this would work perfectly well although the class would require its own code for calculating whether a cluster is allocated/unallocated/end of file/damaged. To improve the encapsulation of the system I will create a separate file allocation table class which will contain methods that do this checking so the analyser class only needs to use methods such as isDamaged().

The file allocation table class will have the structure shown in figure 29.

The final data structures used by the FAT file system are directory entries which exist for

File Allocation Table	
int[] entries ArrayList allocated ArrayList unallocated	
boolean boolean boolean ArrayList ArrayList int	isUnallocated() isEndOfFile() isDamaged() getAllocated() getUnallocated getEntry(int e)

Figure 29: Details of the file allocation table classes

every file and directory and contain information about the file it represents such as its size, name and the starting cluster for its data. When searching for a file with a given name the directory entries are the structures which will need to be interrogated. There are two types of directory entry; standard entries are used for all files and are used in isolation with files whose names are less than 8 characters, long file name entries are used when the file name is greater than 8 characters and occur before the standard directory entry. As these elements' internal structures are almost completely different there would be little point in having a directory entry superclass and using inheritance. Instead 2 separate classes will be used to represent these structures; a FATDirectoryEntry class for standard entrys and a FATLFNDirectoryEntry class for long file names.

FATDirectoryEntry	FATLFNDirectoryEntry
char firstChar String chars2to11 int attributes int createdTimeTenthSeconds int createdTimeHrsMinsSecs int createdDay int accessedDay int highClusterAddress int writtenTimeHrsMinsSecs int writtenDay int lowClusterAddress int fileSize	int sequenceNumber String chars1to5 int attributes int checksum String chars6to11 String chars12to13
char getFirstChar() String getChars2to11() int getAttributes() int getCreatedTimeTenthSecs() int getCreatedTimeHrsMinsSecs() int getCreatedDay() int getAccessedDay() int getHighClusterAddress() int getWrittenTimeTenthSecs() int getWrittenDay() int getLowClusterAddress() int getFileSize() String toString()	int getSequenceNumber() String getChars1to5() int getAttributes() int getChecksum() String getChars6to11() String getChars12to13() String toString()

Figure 30: Details of the Directory Entry classes

All of these data structures will be used by an analyser class which has methods to perform all the required functions such as searching files. Because each file system uses different data structures there will need to be a different analyser class for each file system. As the implementation of the methods will be significantly different for each class I have decided against creating a base analyser class and using inheritance as this would provide no advantages for the implementation and the base class would never be used. As the system being developed is only an example system and because the disk to be used during tested uses a FAT16 file system I have chosen to concentrate on the FAT16 analyser class; the details of which are given below:

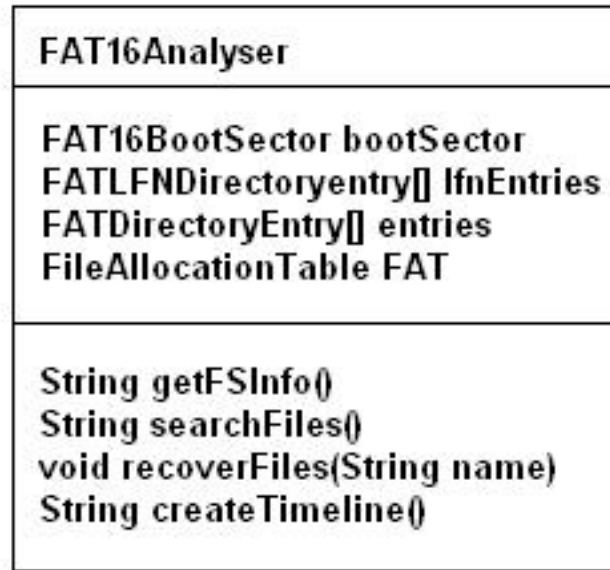


Figure 31: Details of the FAT16 Analyser class

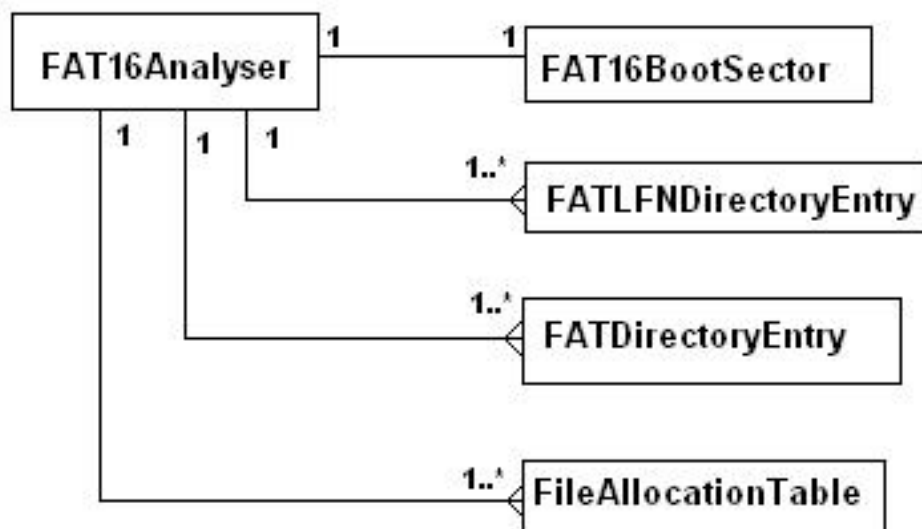


Figure 32: Relationship between the FAT16 File System Analysis Tool classes

5 Implementation

In this section I will highlight the significant problems faced during the implementation phase and the solutions adopted to overcome them.

5.1 Disk Imaging Tool

The standard Java APIs don't provide any methods to communicate directly with the hard disk or other system resources, it is only possible to create, read from and write to files and the disk can't be considered a file. It was therefore necessary to use the C++ language which can manipulate system resources, however there were no in-built functions capable of sending ATA commands to the disk controller and this would be necessary to be able to detect host protected areas and device configuration overlays.

The only solution that seemed reasonable was to write the disk imaging tool in assembly language which could then use the appropriate system interrupt routines to communicate with the disk controller. After testing some assembly code, available on the internet, that supposedly performed the tasks required and reading messages in a specialist forum it became clear that Microsoft Windows versions NT and greater wouldn't allow any program (except itself) to have direct access to system resources. There were only two options available; I could write an assembly program that performed the required tasks providing that it was run on a system using a different operating system to modern versions of windows. This would also be possible by creating a bootable CD which could be used to run the program. Alternatively, Microsoft provides a number of C/C++ APIs that contain functions for limited disk manipulation which I could use to provide some of the necessary functionality. The advantage of using these APIs is that there are in-built functions that gather the information about the disk which would be required by some of my systems functions and reading directly from the disk becomes relatively straight forward. However there are no API commands for sending ATA commands to the disk controller so this approach wouldn't allow some of the requirements to be met. In the interest of keeping the implementation as simple as possible and due to the limited amount of time available I chose to use the Microsoft APIs.

Microsoft's MSDN website provides example code that gathers disk information and this was used for the major part of the 'get disk info' function although some code needed to be added to write the data to a file. The imaging function required this data to be processed from the file so that the size of the disk, and number of bytes per sector could be found. However the problem here, due to my lack of experiencing using C++, was to convert the text representation of the numbers (e.g. "512") to an actual integer. This process is simple using C but the disk access APIs would only work with C++ and the C++ examples found were quite complex and would have taken time to work through and understand. Instead I managed to find a library that would allow C functions to be used in C++ programs.

As already mentioned, the functions for reading bytes directly from the disk were simple to use and didn't cause any problems. However it was necessary to allow the user to choose how many sectors were to be read at a time and it was easily possible that the number of sectors chosen wouldn't be equally divisible by the total number of sectors in which case an error would occur during the final read operation, e.g. if the program attempted to read 20 sectors from the final 12 sectors on the disk. The solution here was to use integer division to calculate how many read operations could be successfully performed given the number of sectors per read and then calculate how many sectors would be left to read by using the modulus (%) operator to find the remainder of the division.

If the chosen number of sectors to read at a time was too large then the system would run out of memory and crash, because these C++ programs would eventually be accessed from the main java interface it was decided to handle invalid values from that interface rather than in the C++ program.

Once both the C++ programs were completed they were tested by attempting to image a USB storage device with a capacity of 1GB (1,028,600,000 bytes) and the program was successful in creating an image. However when I attempted to create an image of a larger storage device (40GB) only a 2GB (actually 1.999...GB) image was created. The problem was something not often considered when programming - a signed 32 bit integer was being used to store the disk size and the maximum value this integer can represent is 2147483647 (approximately 2GB). The first solution I attempted was to use a long integer although it was discovered that they also use 32 bits to store the value, instead it was necessary to use doubles, which store the value in 64 bits,

to store the disk size although doing so had consequences for some of the calculations used. The modulus operator can't be used with double values so it was necessary to change the code that calculates how many sectors are to be read in the final read operation.

The size of the disk in bytes was divided by the number of bytes to be read at a time, this value was truncated and multiplied by the number of bytes to be read at a time to find the total number of bytes that could be successfully read. This value was subtracted from the size of the disk to find how many bytes would be read by the final read operation and this was divided by the number of bytes per sector. This final number of sectors to read was then small enough to be stored in an integer as before. The code to do this is shown below:

```
int sectorSize = getBytesPerSector(infoFile);
double diskSize = getDiskSize(infoFile);
int remainder=0;
int sectorsRead = (diskSize / sectorSize) / sectorsPerRead;
double remainderd = trunc(diskSize / (sectorsPerRead * sectorSize)) *
    (sectorsPerRead * sectorSize);
remainderd = (diskSize - remainderd) / sectorSize;
remainder = remainderd;
int bytesRead = (diskSize / (sectorSize * sectorsPerRead))*(sectorSize*sectorsPerRead);
bytesRead = bytesRead + (remainder*sectorSize);
```

Using a 64-bit double to store the data isn't a limitation at the moment as the maximum value is equal to the (current) maximum size of a hard disk - approximately 2TB - however it does mean that the imaging program will need to be rewritten when hard disk sizes increases and larger disks are commonly encountered.

Once the modifications described above had been completed I again attempted to image a 40GB drive and this time only a 4GB image was created. After checking the program for errors it was discovered that the problem lay in the file system my computer was using. FAT file systems record the size of every file created and this value is stored in an unsigned 32-bit integer - the maximum value of which is approximately 4GB. The only two solutions I considered were to alter the program so that it divided the size of the disk by 4GB and then create the image over that number of files, each up to 4GB in length. However this would add complexity to the system which, when locating data in a particular sector, would need to determine which of the image files the data was located in and then calculate the data's position in that file. As the amount of time available to complete this part of the system was decreasing I decided to assume that the user would be using a file system that didn't impose a restriction on the maximum file size, such as NTFS.

When using the tool to image a drive the user needs to enter the path of the device to be imaged, however if they enter a drive letter such as "C:" then only the partition containing a file system with volume label C will be imaged. To image an entire disk the user has to enter "PhysicalDrive0", "PhysicalDrive1", etc. or whatever the system uses to identify each physical device. I investigated whether there was any way of translating a volume label into a device identifier but there were no obvious solutions so I decided to assume that the user would know the identifier for the device they wish to image.

When integrating the disk imaging program into the main Java interface it became necessary to identify when an error occurs in the imaging program so that the Java program could produce an appropriate warning. This was fine for the program that retrieves disk information as it could be made to return different values depending on what error occurred or if the execution was successful. The Java program could wait for the C++ program to finish its execution, as the process takes less than a second, and then process the result. However the imaging program could take several hours or even days to complete its task depending on the size of the disk being imaged. Forcing the java program to suspend its activity and wait for the imaging tool to return a value wasn't an option as the user may want to use other features of the toolkit whilst the process is

running in the background. The only option was to make sure that the imaging program didn't cause an error and this was done by making a test program that accepted the same inputs as the imaging program, performed the same basic calculations and accessed the same files/drives but didn't do anything else. This program would finish its execution within seconds regardless of success or failure and if the program returned an error value then it would be known that the imaging program would cause the same error so the Java program would produce a warning and avoid running the imaging program. If the test program returned a success value then the imaging program could be used successfully with those inputs.

It was reasonable to assume that if the program will take hours or days to complete a task then the user would want access to a measure of its progress and would also want the option of cancelling the task. Cancelling the imaging process was simple, it can be done by destroying the thread in which the imaging process is running, however producing a progress bar was more complicated. The problem was that whilst the imaging program could generate values that represent how much progress has been made the Java program can't access those values until the process terminated. Two possible solutions were to continue experimenting with using java to handle output from the imaging program or to create a progress bar from within the imaging program. However I didn't have the necessary time to experiment with thread IO in java and I didn't have any experience in GUI programming in C++ to be able to make a progress bar.

When I had been creating images during testing using windows explorer to view the properties of the disk being imaged caused an access violation exception that I'd found no way of handling. Because of this I'd assumed that a program couldn't access the disk being imaged during the process. However after some experimentation I found that it was possible to create a Java file object that represented the image file being created and to find the number of bytes in that file. Doing this made it possible to use the size of the disk, which could be obtained from the file created by the program that gathers disk information, as the maximum value of the progress bar and use the size of the image file as the current value of the progress bar. The only problem was that the JProgressBar provided by swing can only use integers for its values which meant that the size of most disks couldn't be stored. I decided to store the values I was going to use as doubles, divide them both by 1,000,000 and then store them as integers which could be used by the progress bar and would provide a reasonable approximation to the amount of progress made.

5.2 Partition Analysis Tool

Compared to the disk imaging tool this part of the project was relatively simple although there were a couple of problems.

The first, and simplest, problem was to do with the swing implementation of the interface for the forms that gathered partition information. The form was divided into two panels; one panel contained the text boxes and buttons for the user input and the other panel contained a text area which would display the results if requested. The problem was that the text area for the results was out of the scope of the action listener that created an instance of a dos partition analyser that retrieved the partition information. To overcome this problem it was necessary to have the action listener class located in the main class for the form and have a method in the panel class which could add the action listener to the appropriate button. The action listener could then use modifier methods of the other panel class to write to the text area.

The more serious problem was that during testing the program would return data which clearly wasn't from a dos partition table; for example it would indicate that none of the partitions were bootable or it would return partition types that weren't recognised. The cause of this problem was the method of converting the bytes read from the image file into a string of 1s and 0s that represented the bytes, e.g. reading 00001111 and converting to "00001111". I had assumed that the toString() method in the Byte class carried out the required conversion, however this wasn't the case it would instead convert the byte into an integer and then return this integer as a string. Instead the program needed to convert the bytes into this string provided by the Byte.toString() method and use this string in the constructor of an Integer object, the int value of this Integer class was then used by a separate program I created that will convert an integer into the desired

string of 1s and 0s. It was necessary to obtain a string of this form for each of the data structures used so that different bytes could be extracted (e.g. in the dos boot sector the partition table is located in bytes 446-511) and used in the analysis.

5.3 File System Analysis Tool

Most of the problems that were likely to occur had already been solved when implementing the partition analysis tool, e.g. how to correctly convert bytes into string representations. However one problem I faced was not knowing which fields of the data structures are stored in little endian format, in which case the bytes would need to be reversed before being translated to integers or strings, and which are stored in big endian format. If the wrong scheme was used then the data returned by the tool would obviously be incorrect, e.g. a cluster address of “0000000000000001” could be converted to 1 in big endian or 256 in little endian, and the only way to solve this problem was to use one addressing scheme for each field, attempt to analyse an image file and check whether the results were correct and if not the other addressing scheme should be used. This problem didn’t arise with the partition analysis tool as the documentation used clearly stated how each field in the partition tables are stored.

6 Testing

In this section I will describe the tests that were carried out on each tool, the results of the tests and how the test suite was created using the category partition method

6.1 Disk Imaging Tool

6.1.1 Test Specification

The ‘Get Disk Information’ function’s parameters are the path of the disk (or partition) to image, the path of the information file to create and whether the information should be displayed on the screen. From this the categories identified are;

- The validity of the disk path
- The validity of the information file path
- Whether the information should be displayed on screen.

The equivalence class/partitions are therefore:

The validity of the disk path:

The disk path is valid

The disk path is not valid

The validity of the information file path:

The information path is valid

The information path is not valid

Whether the Information should be displayed on screen:

The information should be displayed

The information should not be displayed

The ‘Create Disk Image’ function’s parameters are; the path of the target disk, the path of the image file to be created, the path of the log file to be generated, the number of sectors to be read per operation and the path of the file containing the disk information.

The categories identified are:

- The validity of the disk path
- The validity of the image file path
- The validity of the log file path
- The validity of the number of sectors per read
- The existence of the information file
- The validity of the information file

The validity of the number of sectors per read is included to test for cases where the user enters data other than all digits (e.g. “2t3”) and this category also includes the range of values entered. The validity of the information file is a separate category to its existence because the function requires that the file has a certain format and it would cause an error if this wasn’t the case.

The partitions which will be used are:

The validity of the disk path:

The disk path is valid

The disk path is not valid

The validity of the image file path:

The image file path is valid

```

    The image file path is not valid
The validity of the number of sectors per read:
    The number is not valid
    The number is less than 1
    The number is greater than 300
    The size of the disk is divisible by the value
    The size of the disk isn't divisible by the value
The existence of the information file:
    The file exists
    The file doesn't exist
The validity of the information file:
    The information file is not valid
    The information file is valid

```

There is no scientific reason for the maximum number of sectors being 300 but there is technical limitation to the amount of memory the program can use so 300 sectors (150KB for a sector size of 512 bytes) is essentially a randomly chosen figure. Quite a low maximum was chosen because a read error causes the bytes in all the sectors that were to be copied to be lost - in the sense that they're not copied to the image file. Cases where the size of the disk needs is/isn't divisible by the number by the number of sectors per read also need to be tested to make sure that the entire disk is copied in each case.

The cases where the disk path, image path or information file has been marked as an error because the program couldn't function if any of these properties hold and testing them in combination with other values probably wouldn't reveal any extra faults than testing them once.

Whilst it isn't a major requirements it is likely that the user will want to view the information file from within the program rather than having to use another text editor so it is important that the 'View Disk Information' function is thoroughly tested. This function has only one parameter; the path of the information file to open.

The categories identified for this parameter are:

- The existence of the information file
- The validity of the information file

and from this the partitions can be identified as:

```

Information File existence:
    File does exist
    File doesn't exist
Information File validity
    File is valid
    File is not valid

```

The file being invalid is not considered to be an error in the context of the specification because attempting to open an invalid file will not cause a serious system error and so can be tested with combinations of inputs.

The final function is the 'View Imaging Log' function which also takes only one parameter - the path of the log file. This function will have the following categories:

- Log File existence
- Log File validity

and these can be partitioned in the following way:

Log File Existence:
 Log file exists
 Log file doesn't exist
Log File Validity
 File is valid
 File is not valid

These partitions can be represented in the following specification:

6.1.2 Test Frames

At the time of testing I did not have access to the catpart testing tool, however I was able to manually generate the following test frames from the test specifications.

A: Get-Disk-Information

- 1 disk-path-validity = invalid
- 2 information-path-validity = invalid
- 3 disk-path-validity = valid
 information-path-validity = valid
 display-information = yes
- 4 disk-path-validity = valid
 information-path-validity = valid
 display-information = no

B: Create-Disk-Image

- 5 disk-path-validity = disk path is not valid
- 6 image-path-validity = image path is not valid
- 7 information-file-existence = file doesn't exist
- 8 information-file-validity = file is not valid
- 9 disk-path-validity = disk path is valid
 image-path-validity = image path is valid
 validity-of-sector-per-read = Value is invalid
 information-file-existence = file exists
 information-file-validity = file is valid
- 10 disk-path-validity = disk path is valid
 image-path-validity = image path is valid
 validity-of-sector-per-read = Value < 1
 information-file-existence = file exists
 information-file-validity = file is valid
- 11 disk-path-validity = disk path is valid
 image-path-validity = image path is valid
 validity-of-sector-per-read = Value > 300
 information-file-existence = file exists

information-file-validity = file is valid

- 12 disk-path-validity = disk path is valid
image-path-validity = image path is valid
validity-of-sector-per-read = Value is between 1 and 300
information-file-existence = file exists
information-file-validity = file is valid
- 13 disk-path-validity = disk path is valid
image-path-validity = image path is valid
validity-of-sector-per-read = Size of disk is not equally divisible
information-file-existence = file exists
information-file-validity = file is valid
- 14 disk-path-validity = disk path is valid
image-path-validity = image path is valid
validity-of-sector-per-read = Size of disk is equally divisible
information-file-existence = file exists
information-file-validity = file is valid

Due to space limitations the remaining test frames can be found in Appendix B

6.1.3 Results

Test Number	Test Frame	Disk Path	Information File	Display Information	Expected Outcome	Actual Outcome
1	1	Physical Drive	G: USB Info.txt	Yes	An error message should be displayed	An error message was displayed
2	2	PhysicalDrive2	G	Yes	An error message should be displayed	An error message was displayed
3	3	PhysicalDrive2	G:USB Info.txt	Yes	The information should be gathered and displayed	The information was gathered and displayed
4	4	PhysicalDrive2	G:USB Info.txt	No	The information should be gathered	The information was successfully gathered

Below are the results of testing the 'Create Disk Image Function'

Test Number	Test Frame	Disk Path	Image Path	Sectors Per Read	Information File	Expected Outcome	Actual Outcome
5	5	phy	G:Image.dat	20	G:Info.txt	An error message should be displayed	An error message was displayed
6	6	Physical Drive2	Ima	20	G:Info.txt	An error message should be displayed	An error message was displayed
7	7	Physical Drive2	G:Image.dat	20	G:NotHere.txt	An error message should be displayed	An error message was displayed
8	8	Physical Drive2	G:Image.dat	20	G:Invalid.txt	An error message should be displayed	An error message was displayed
9	9	Physical Drive2	G:Image.dat	t	G:Info.txt	An error message should be displayed	An error message was displayed
10	10	Physical Drive2	G:Image.dat	-2	G:Info.txt	An error message should be displayed	An error message was displayed
11	11	Physical Drive2	G:Image.dat	301	G:Info.txt	An error message should be produced	An error message was displayed
12	12	Physical Drive2	G:Image.dat	10	G:Info.txt	The disk should be successfully imaged	The disk was successfully imaged
13	13	Physical Drive2	G:Image.dat	11	G:Info.txt	The disk should be successfully imaged	The disk was successfully imaged
14	14	Physical Drive2	G:Image.dat	1	G:Info.txt	The disk should be successfully imaged	The disk was successfully imaged

The table below contains the result of testing the 'View Disk Information' function:

Test Number	Test Frame	Information File	Expected Outcome	Actual Outcome
15	15	G:Not Exist.txt	An error message should be displayed	An error message was displayed
16	16	G:Info.txt	The file should be displayed	The file was displayed
17	17	G:Invalid.txt	The file should be displayed	The file was displayed

The table below shows the results of testing the 'View Imaging Log' function.

Test Number	Test Frame	Log File	Expected Outcome	Actual Outcome
18	18	G:Not Exist.txt	An error message should be displayed	An error message was displayed
19	19	G:Log.txt	The file should be displayed	The file was displayed
Test Number	Test Frame	Information File	Expected Outcome	Actual Outcome

6.2 Partition Analysis Tool

6.2.1 Test Specification

The ‘get partition information’ function has the following parameters; the path of the image file to analyse, the path of the results file to be created, the path of the log file to be created, the path of the information file to be used and whether the results should be displayed.

These parameters can be organised into the following categories:

- image file existence
- image file validity
- results file path validity
- log file path validity
- information file existence
- information file validity
- results to be displayed

These categories can be organised into the following partitions:

```
Image File Existence
  Image file exists
  Image file doesn't exist
Image File Validity
  Image file is valid
  Image file is not valid
Results File Path Validity
  Results file path is valid
  Results file path is not valid
Log File Path Validity
  Log file path is valid
  Log file path is not valid
Information File Existence
  File exists
  File doesn't exist
Information File Validity
  File is valid
  File isn't valid
Results to be displayed?
  Yes
  No
```

The cases where the image file doesn't exist, the results file path isn't valid, the log file path isn't valid, the information file doesn't exist and the information file isn't valid are specified as errors because these events would stop the program from being able to obtain continue analysing the data. As a result these cases only need to be tested once because they should produce the same result regardless of what other parameters are entered.

The ‘view partition information’ function only has one parameter - the path of the results file. This parameter has the following two categories:

- Results File Existence
- Results File Validity

which can be organised into the partitions given below:

Results File Existence

File exists

File doesn't exist

Results File Validity

File is valid

File is not valid

these partitions can be described in the following specification:

6.2.2 Test Frames

See Appendix B

6.2.3 Results

The table below contains the results of the testing on the 'get partition information' function.

Test Number	Test Frame	Image File	Results File	Log File	Information File	Display Results	Expected Outcome	Actual Outcome
1	21	G:Not Ex-ist.dat	G:Res.txt	G:Log.txt	G:Info.txt	Yes	An error message should be displayed	An error message was displayed
2	22	G:Image .dat	file	G:Log.txt	G:Info.txt	Yes	An error message should be displayed	An error message was displayed
3	23	G:Image .dat	G:Res.txt	file	G:Info.txt	Yes	An error message should be displayed	An error message was displayed
4	24	G:Image .dat	G:Res.txt	G:Log.txt	file	Yes	An error message should be displayed	An error message was displayed
5	25	G:Image .dat	G:Res.txt	G:Log.txt	G:Invalid .txt	Yes	An error message should be displayed	An error message was displayed
6	26	G:Image .dat	G:Res.txt	G:Log.txt	G:Info.txt	Yes	The data should be gathered and displayed	The data was gathered and displayed
7	27	G:Image .dat	G:Res.txt	G:Log.txt	G:Info.txt	No	The data should be successfully gathered	The data was successfully gathered
8	28	G:Invalid .dat	G:Res.txt	G:Log.txt	G:Info.txt	Yes	Incorrect data should be displayed	Incorrect data was displayed
9	29	G:Invalid .dat	G:Res.txt	G:Log.txt	G:Info.txt	No	Incorrect data should be gathered	Incorrect data was gathered

The table below contains the result of testing the 'view partition information' function.

Test Number	Test Frame	Information File	Expected Outcome	Actual Outcome
1	30	G:Not Exist.txt	An error message should be displayed	An error message was displayed
2	31	G:Partition Info.txt	The information should be displayed	The information was displayed
3	32	G:Invalid Partition Info.txt	The contents of the file should be displayed (in ascii)	The contents of the file were displayed in ascii format

6.3 File System Analysis Tool

6.3.1 Test Specification

The ‘load partition’ function is used on the interfaces for every file system analysis function and was therefore tested separately. This function takes the following parameters; the image file, the partition information file and the disk information file. These parameters can be divided into the following categories:

- Image File Existence
- Partition Information File Existence
- Partition Information File Validity
- Disk Information File Existence
- Disk Information File Validity

and the following parameters can be identified from these categories.

Image File Existence:

File exists

File doesn’t exist

Partition Information File Existence:

File exists

File doesn’t exist

Partition Information File Validity:

File is valid

File is not valid

Disk Information File Existence:

File exists

File doesn’t exist

Disk Information File Validity

File is valid

File is not valid

The ‘get file system information’ function has the following parameters; the partition to be analysed, the path of the results file, the path of the log file and whether the information is to be displayed on the screen.

These parameters can be organised into the following categories;

- Has a partition been selected
- Is the partition type supported
- The validity of the results file path
- The log file path validity
- Is the information to be displayed

For the purpose of testing the type of a partition is considered to be the type of file system, or other system (e.g. a partition used for system recovery) it contains. A file system not being supported includes the situations where a criminal may have corrupted the file system structures to the extent where the system can't be analysed. Also, since the 'load partition' function is always executed before this function it has been assumed that the parameters used by that function will be valid if the program reaches the point of execution where this function is called. The categories were organised into the partitions shown below:

```
Has a partition been Selected?
  Yes
  No
Is the partition type supported?
  Yes
  No
Validity of the results file path
  Path is valid
  Path is not valid.
Log file path validity
  Path is valid
  Path is not valid
Display Information
  Yes
  No
```

The 'search files' function has 7 parameters; the partition to be analysed, the type of search to be conducted (e.g. search file names), the file types to be searched (e.g. .jpg, .doc etc.), the status of files to be searched (e.g. search deleted files), the search criteria, whether all or some of the terms should match and the path of the results file to create.

These parameters can be organised into the following categories;

- Has a partition been selected
- Whether the partition type is supported
- Has a search type been selected
- The search type
- The file types to be searched
- Whether a file status has been selected
- The file status selected
- The search criteria
- How many terms must match for a result
- The validity of the results file path

These categories can be used to create the following partitions:

```
Has a partition been selected:
  Yes
  No
Is the partition type supported:
  Yes
  No
```

Has a search type been selected:

Yes

No

The search type selected:

File Name

File Content

Both

The File types to be searched:

Images (.jpg, .gif, .png, .bmp)

Text (.doc, .rtf, .txt)

JPEG

GIF

Bitmap

PNG

Word document

Text document

Rich Text Format

All

Has a file status been selected:

Yes

No

The file status selected:

Deleted

Current

Hidden

All

The search Criteria

Empty String

Non-Empty String

The validity of the results file path:

valid path

invalid path

The 'recover files' function has three parameters; the partition to be analysed, the name of the file to be recovered and the path of the file to recover the file to. These parameters can be used to create the following categories:

- Has a partition been selected
- Is the partition type supported
- The existence of the file to recover
- The validity of the recovered file path

these categories can be organised into the following partitions:

Has a partition been selected:

Yes

No

Is the partition type supported:

Yes

No

The existence of the file to recover:

The file exists

The file doesn't exist

The validity of the recovered file path

The file path is valid

The file path is not valid

The test specification for these partitions and the test information for the ‘create timeline’ function can be found in appendix B.

6.3.2 Test Frames

See Appendix B.

6.3.3 Results

The table below contains the results of testing the ‘load partition’ function

Test Number	Test Frame	Image File	Partition Information File	Disk Information File	Expected Outcome	Actual Outcome
1	33	G:Not Exists .dat	G:Partition Info.txt	G: Disk Info .txt	An error message should be displayed	An error message was displayed
2	34	G:Image .dat	G: Not Exist .txt	G: Disk Info .txt	An error message should be displayed	An error message was displayed
3	35	G:Image .dat	G: Invalid.txt	G: Disk Info .txt	An error message should be displayed	An error message was displayed
4	36	G:Image .dat	G: Partition Info.txt	G:Not Exist .txt	An error message should be displayed	An error message was displayed
5	37	G:Image .dat	G: Partition Info.txt	G:Invalid .txt	An error message should be displayed	An error message was displayed
6	38	G:Image .dat	G: Partition Info.txt	G: Disk Info .txt	The partition information should be loaded	The partition information was loaded

Due to space limitations the test results of testing the ‘search files’ function can be found in appendix B.

The following table contains the results of testing the ‘recover files’ function:

Test Number	Test Frame	Partition to be Analysed	File to recover	Recovered File Path	Expected Outcome	Actual Outcome
1	279	None selected	recover.txt	G: recover.txt	An error message should be displayed	An error message was displayed
2	280	1. NTFS (Hidden)	recover.txt	G: recover.txt	An error message should be displayed	An error message was displayed
3	281	2. FAT16	recover.txt	G: recover.txt	The file should be recovered	The file was recovered
4	282	2. FAT16	recover.txt	Invalid	An error message should be displayed	An error message was displayed
5	283	2. FAT16	doesn't exist.txt	G: recover.txt	A message should be displayed informing the user that the file doesn't exist	A message was displayed informing the user that the file doesn't exist
6	284	2. FAT16	doesn't exist.txt	Invalid	An error message should be displayed	An error message was displayed

7 Conclusions

7.1 Summary

The rise in computer usage for creation of personal/business documents and financial transactions has lead to a major increase in high-tech crimes such as identify fraud. These crimes require an advanced knowledge of how computers are organised internally along with forensic techniques of recovering evidence of these crimes, evidence that a criminally may have attempted to destroy.

Evidence can be hidden without any regard to the structures used by the operating system and other software by placing the data into an unused area of the disk. Access to this area of the disk can be restricted by creating a host protected area, or device configuration overlay, or alternatively an already restricted area such as bad sectors can be used to store the data. However these hiding places can be easily identified and analysed. Methods of wiping the disk to destroy evidence have also been discussed along with the limitations of zero-footprinting software.

The basic concepts of volumes and partitions were discussed, with a particular focus on DOS partition systems, along with the general locations in which data can be hidden such as partition slack. More specific examples relating to the details of particular partitioning systems were examined, such as setting a DOS partition type as hidden or hiding data in unused entries of a GUID partition table.

I have explained the details of how different file systems store and delete files, with a particular emphasis on NTFS file systems, and places where files may still reside after they've been deleted. As part of this file recovery techniques such as simply reading unallocated clusters were discussed along with their limitations. Data Carving is an important part of computer forensics in that it allows fragmented files to be located and recovered that wouldn't be possible by other methods. In the event that data is hidden rather than deleted locations in which data can be hidden, such as reserved sectors or bad clusters, were highlighted and would be examined during a digital investigation.

Data isn't always hidden, a criminal may often use encryption to hide the true content of the data rather than its existence. Breaking strong encryption by use of a brute-force attack is often infeasible in terms of the amount of time it would take. However there are other factors which may get around encryption such as the suspect writing down their password which can then be used to find the encryption key. Alternatively, unencrypted versions that the user is unaware of

may exist elsewhere on the disk.

It isn't always the case that the documents are used in a crime, sometimes its the computer activity which is illegal, for example visiting an online gambling site in the US. Basic techniques for event reconstruction using the time stamps of files have been discussed along with analysis of internet activity by examining the history files created by browsers.

Finally, modern techniques involve hiding data within another file in such a way that it is undetectable by simply viewing the original file. This techniques is called steganography and a number of methods for hiding data within images and audio files, such as least significant bit encoding, have been discussed along with some potential methods for detecting their use.

This report also briefly examined some of the current computer forensics toolkits that are commercially available. These tools are capable of analysing most file system types and some also provide decryption features although they may be prohibitively expensive for smaller organisations. To date, none of the most commonly used tools are capable of steganalysis.

7.2 Work Achieved

As part of this project I managed to analyse many of the techniques used by computer forensics analysts. Using some of these techniques I was able to build a demonstration application that is capable of analysing certain partition and file systems to recover some of the deleted/hidden files.

This application follows some of the recommended guidelines of digital investigations by creating copies of the original disk and making logs of the analysis performed.

7.3 Suggestions for Future Work

The first step for a future project would be to expand the application created for this project so that it can analyse a wider range of systems using the techniques identified as part of this project. Also, due to this project focus on analysis, I was unable to incorporate effective event reconstruction, encrypted file analysis or steganalysis methods. Future work could also incorporate this functionality into the existing program.

One of the major limitations of this project is that it only described methods of finding hiding locations for evidence but provided no techniques for determining whether any data was hidden in those locations and whether the data hidden there could be considered to be evidence. My suggestions for future projects would be to focus on these areas which are necessary for investigating more advanced crimes, or rather crimes where the criminal is more technically gifted.

Another area I would be interested in for future projects is methods of reconstructing files once all the fragments have been identified. One proposal given in section 2 was to analyse the edges of an image fragment, i.e. the bytes representing the pixel values at the edges of the picture, and then piecing together similar edges like a jigsaw.

The files recovered may need enhancement before they can be used as evidence. One example of this would be cctv footage which may need to be sharpened so that a person's face can be identified, or a sound file may need to be filtered so that one person's voice can be heard clearly. These enhancement techniques can also be used to aid investigations by recovering the fingerprints on a soft surface by analysing the fourier transform of a photograph of the surface. These techniques are worth investigating in future projects.

References

- [Bal04] Craig Ball. Cross Examination of the Computer Forensics Expert. <http://www.craigball.com/expertcross.pdf>, 2004.
- [BGML96] W. Bender, D. Gruhl, N. Morimoto, and A. Lu. Techniques for Data Hiding. *IBM Systems Journal*, 35(3 and 4), 1996. <http://www.research.ibm.com/journal/sj/mit/sectiona/bender.pdf>.
- [BHS] Hal Berghel, David Hoelzer, and Michael Sthultz. Data Hiding Tactics for Windows and Unix File Systems. <http://berghel.net/publications/datahiding/datahiding.php>.
- [Bro] Andries Brouwer. Partition types: Properties of partition tables. http://www.win.tue.nl/~aeb/partitions/partition_types-2.html.
- [Car02] Brian Carrier. An Investigator’s Guide to File System Internals. @Stake, 2002.
- [Car03] Brian Carrier. Defining Digital Forensic Analysis Tools Using Abstraction Layers. *International Journal Of Digital Evidence*, 1(4), 2003. <http://www.utica.edu/academic/institute/ecii/publications/articles/A04C3F91-AFBB-FC13-4A2E0F13203BA980.pdf>.
- [Car05] Brian Carrier. *File System Forensic Analysis*. Addison-Wesley, 2nd edition, June 2005.
- [Cas02] Eoghan Casey. Practical Approaches to Recovering Encrypted Digital Evidence. *International Journal Of Digital Evidence*, 1(3), 2002. <http://www.utica.edu/academic/institute/ecii/publications/articles/A04F2FB-BD97-C28C-7F9F4349043FD3A9.pdf>.
- [CG03] Brian D. Carrier and Joe Grand. A hardware-based memory acquisition procedure for digital investigations. *Digital Investigation Journal*, 1(1):1742 – 2876, December 2003.
- [Chu] Anton Chuvakin. Linux Data Hiding and Recovery. <http://www.linuxsecurity.com/content/view/117638>.
- [CR04] Megan Carney and Marc Rogers. The Trojan Made Me Do It: A First Step In Statistical Based Computer Forensics Event Reconstruction. *International Journal of Digital Evidence*, 2(4), 2004. <http://www.utica.edu/academic/institute/ecii/publications/articles/A0B2CCCB-E6FC-6840-AF4A01356B9B687A.pdf>.
- [CS04] Brian Carrier and Eugene Spafford. Defining Event Reconstruction of Digital Crime Scenes. Technical report, Center for Education and Research in Information Assurance and Security. Purdue University, 2004. http://www.cerias.purdue.edu/tools_and_resources/bibtex_archive/2004-37.pdf.
- [Data] Access Data. Access Product Line – DNA. <http://www.accessdata.com/products/dna/>.
- [Datb] Access Data. Access Product Line – FTK. <http://www.accessdata.com/products/ftk/>.
- [Datc] Access Data. Access Product Line – FTK Imager. <http://www.accessdata.com/products/imager/>.
- [Datd] ASR Data. ASR Data – Smart. <http://www.asrdata.com/SMART/general.html>.
- [Dic06] Daniel Dickerman. Advanced Data Carving. *DFRWS Data Carving Challenge*, 2006.
- [For] ProDiscover Forensics. ProDiscover Forensics – Disk Forensics Tools. <http://www.techpathways.com/ProDiscoverDFT.htm>.

- [GBA⁺93] R.D Gomez, E.R. Burke, A.A. Adly, I.D. MayerGoyz, J.A. Gorczyca, and M.H. Kryder. Microscopic Investigations of Overwritten Data. Technical report, University of Maryland, 1993.
- [Gei05] Matthew Geiger. Evaluating Commercial Counter-Forensic Tools. *Digital Forensic Research Workshop*, 2005. http://www.dfrws.org/2005/proceedings/geiger_couterforensics.pdf.
- [GHR06] Mayank R. Gupta, Michael D. Hoeschele, and Marcus K. Rogers. Hidden Disk Areas: HPA and DCO. *International Journal of Digital Evidence*, 5(1), 2006.
- [HIW] HIW. How it Works Partition Tables. <http://www.ata-atapi.com>.
- [HP98] Gord K. Hama and Mark M. Pollitt. Data Reduction - Refining the Sieve. *Int. Organization on Computer Evidence (IOCE)*, 1998.
- [img] [htt\[://www.dizionarioinformatico.com/IMAGES/disco1.gif](http://www.dizionarioinformatico.com/IMAGES/disco1.gif).
- [JJ98] Neil F. Johnson and Sushil Jajodia. Exploring Steganography: Seeing the Unseen. *Computing Practices*, 1998.
- [Jon] Keith Jones. Forensic Analysis of Internet Explorer Activity Files. http://www.foundstone.com/pdf/wp_index_dat.pdf.
- [JR91] Patrick R. Gallagher JR. A Guide to Understanding Data Remanence in Automated Information Systems. Technical report, National Computer Security Center, 1991.
- [Kre04] Robert Krenn. Steganography and Steganalysis. <http://www.krenn.nl.univ/cry/steg/article.pdf>, March 2004.
- [Kue02] Brian Kuepper. What You Don't See On Your Hard Drive. *SANS Security essentials GSEC Practical Assignment*, 2002.
- [LD] Eugene T. Lin and Edward J. Delp. A Review of Data Hiding in Digital Images. Technical report, Video and Image Processing Laboratory. school of Electrical and Computer Engineering. Purdue University. <ftp://skynet.ecn.purdue.edu/pub/dist/delp/pics99-stego/paper.pdf>.
- [Mar02] Damon Martin. Windows, NTFS and Alternate Data Streams. *SANS Security Essentials GSEC Practical Assignment*, 2002.
- [MR04] Matthew Meyers and Marc Rogers. Computer Forensics: The need for Standardization and Certification. *International Journal of Digital Evidence*, 3(2), 2004. <http://www.utica.edu/academic/institute/ecii/publications/articles/A0B7F51C-D8F9-A0D0-7F387126198F12F6.pdf>.
- [MRMS] S. Mitra, T. Roy, D. Mazumdar, and A.B. Saha. Steganalysis of LSB Encoding in Uncompressed Images By Close Colour Pair Analysis. <http://www.security.iitk.ac.in/IITKHACK04/papers/cp03.pdf>.
- [Sar06] Bryan Sartin. ANTI-Forensics – distorting the evidence. *Computer Fraud and Security*, 2006.
- [SDM⁺] K. Sullivan, O. Dabeer, U. Madhow, B.S. Manjunath, and S. Chandrasekaran. LLRT Based Detection of LSB Hiding. Dept. of Electrical and Computer Engineering. University of California at Santa Barbara <http://vision.ece.ucsb.edu/publications/03ICIPKen.pdf>.
- [SJ00] Tony Sammes and Brian Jenkinson. *Forensic Computing A Practitioner's Guide*. Springer-Verlag London Limited, 5th edition, 2000.

- [SK01] Kimberly Stone and Richard Keightley. Can Computer Investigations Survive Windows XP? Technical report, Guidance Software, 2001. <http://www.encase.com/corporate/downloads/whitepapers/XPwhitepaper.pdf>.
- [Sof06] Guidance Software. EnCase Forensic Detailed Product Description. Technical report, Guidance Software, 2006.
- [SS] Amanda Sharkey and Rod Smallwood. Com 2040/6650 Professional Issues in Information Technology Part VIII: Data Protection, Privacy and the Freedom of Information. Com 2040/6650 Lecture: University of Sheffield.
- [Vac05] John Vacca. *Computer Forensics: Computer Crime Scene Investigation*. Charles River Media Inc, 2nd edition, 2005.
- [Vid05] Arne Vidstrom. Computer Forensics and the ATA Interface. Technical report, FOI, 2005.
- [Wika] Wikipedia. Disk Partitioning Wikipedia the free encyclopedia. <http://www.wikipedia.com>.
- [Wikb] Wikipedia. Volume (Computing). <http://www.wikipedia.com>.
- [Wikc] Wikipedia the free encyclopedia. Steganography. <http://en.wikipedia.org/wiki/Steganography>.

8 Appendix A

8.1 X-Machines

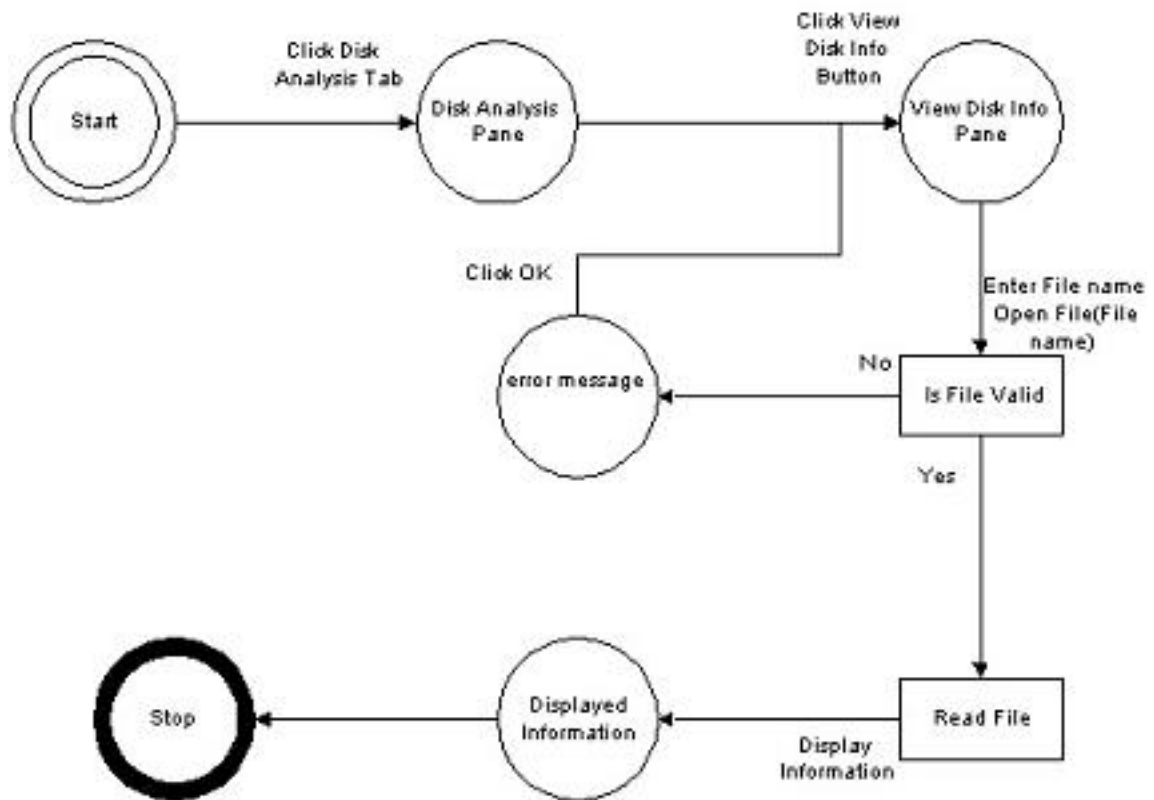


Figure 33: X-Machine for the Get Disk Info Function

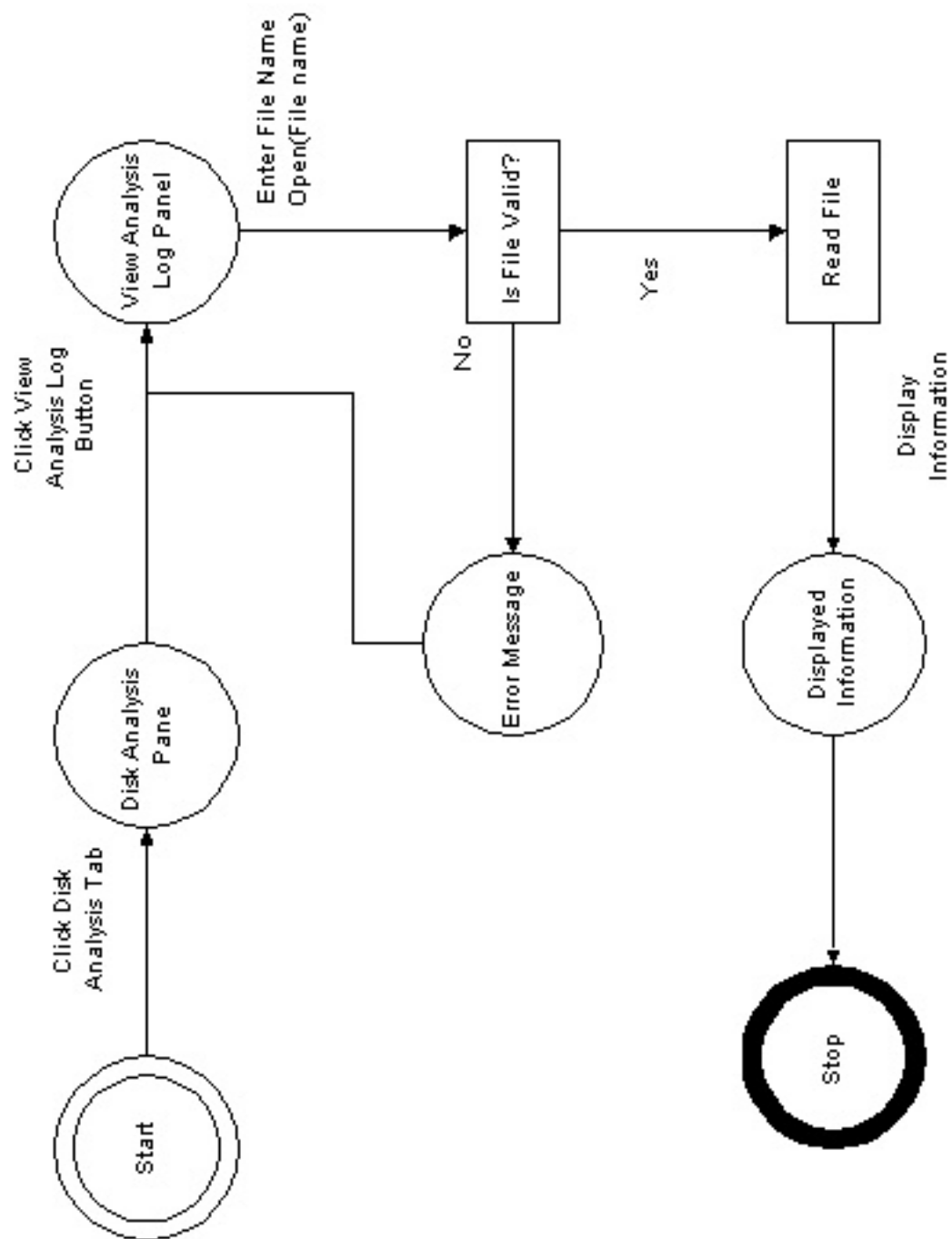


Figure 34: X-Machine for the View Imaging Log Function

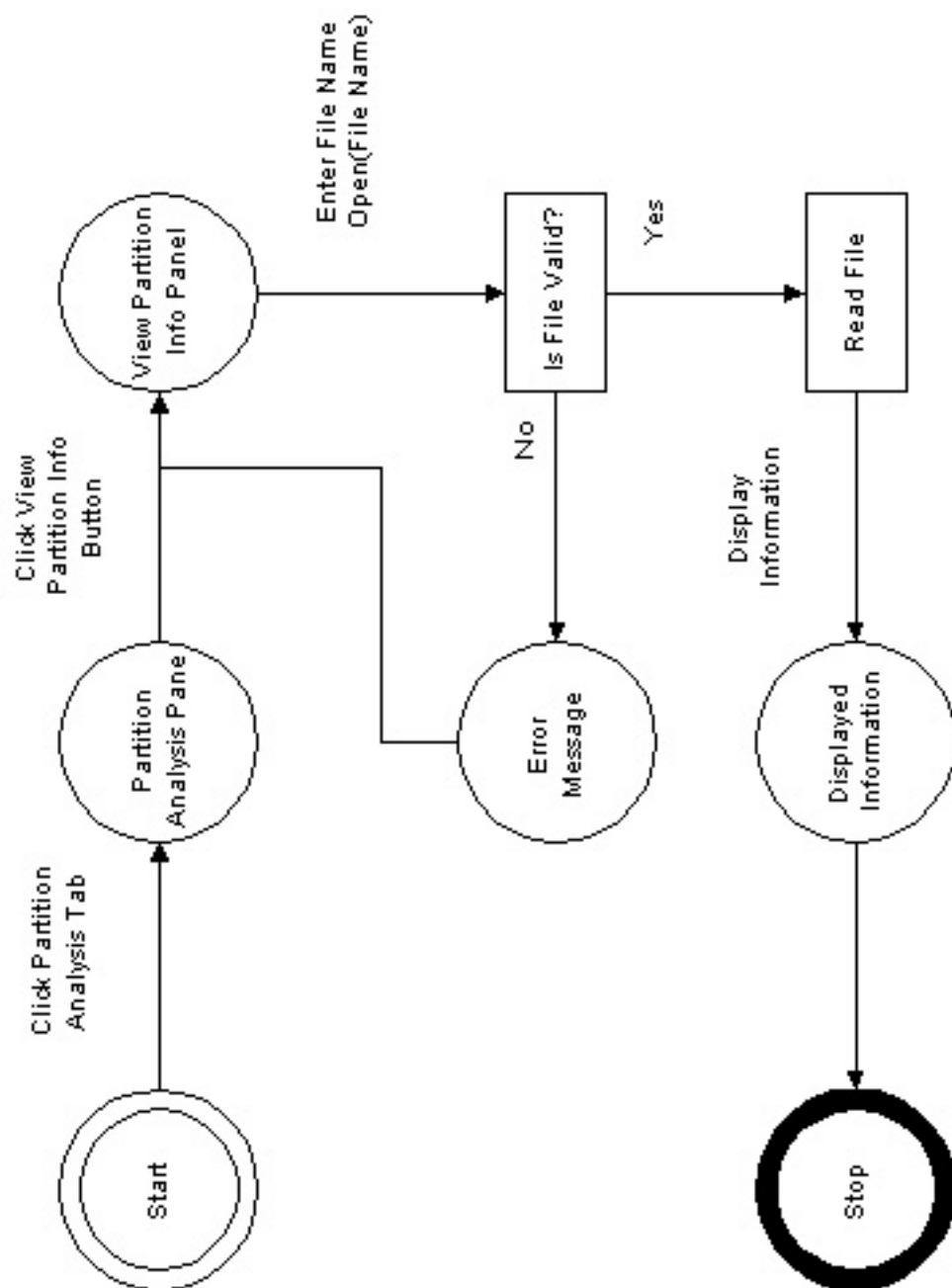


Figure 35: X-Machine for the View Partition Info Function

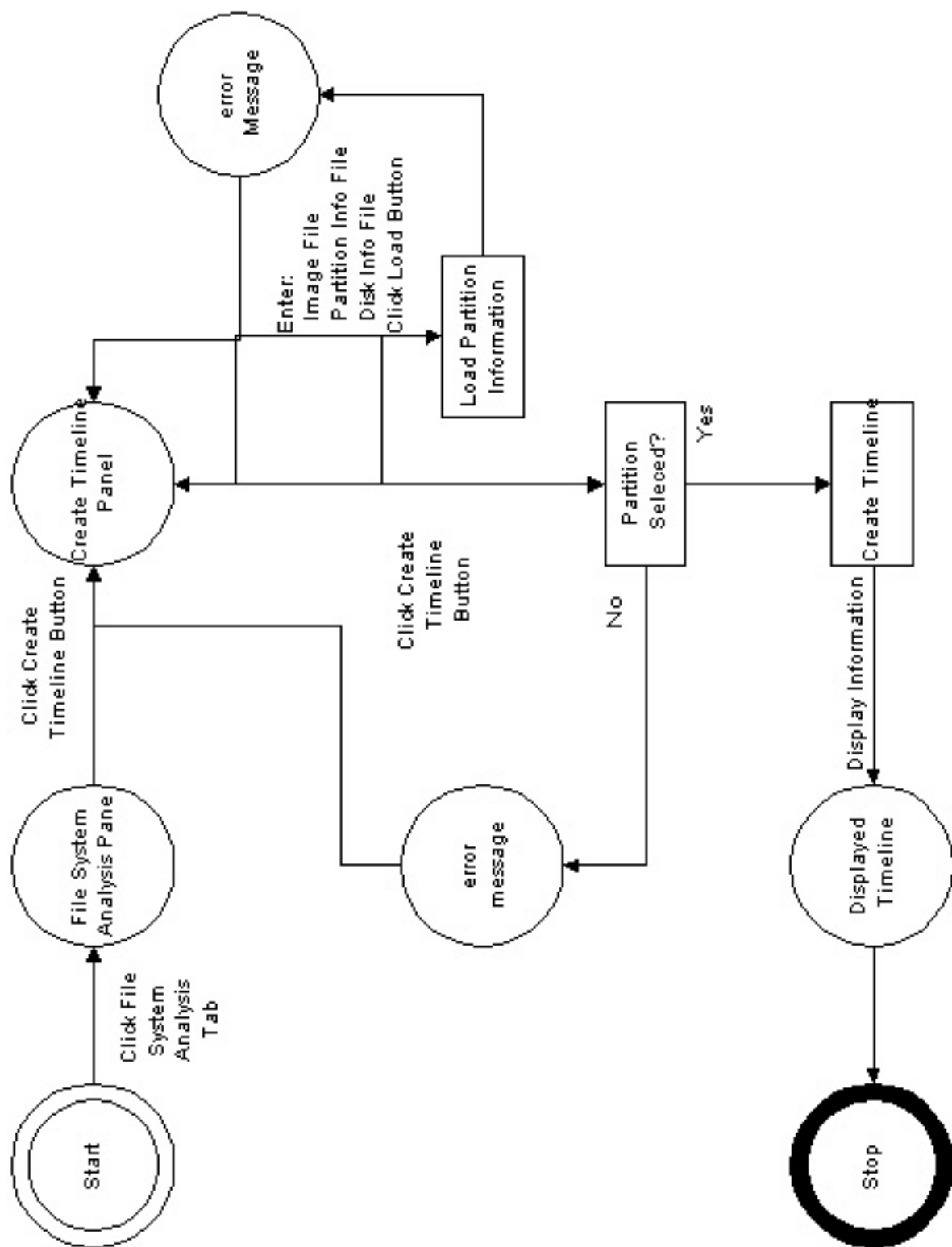


Figure 36: X-Machine for the Create Timeline Function

8.2 Screen-shots of the Prototype System

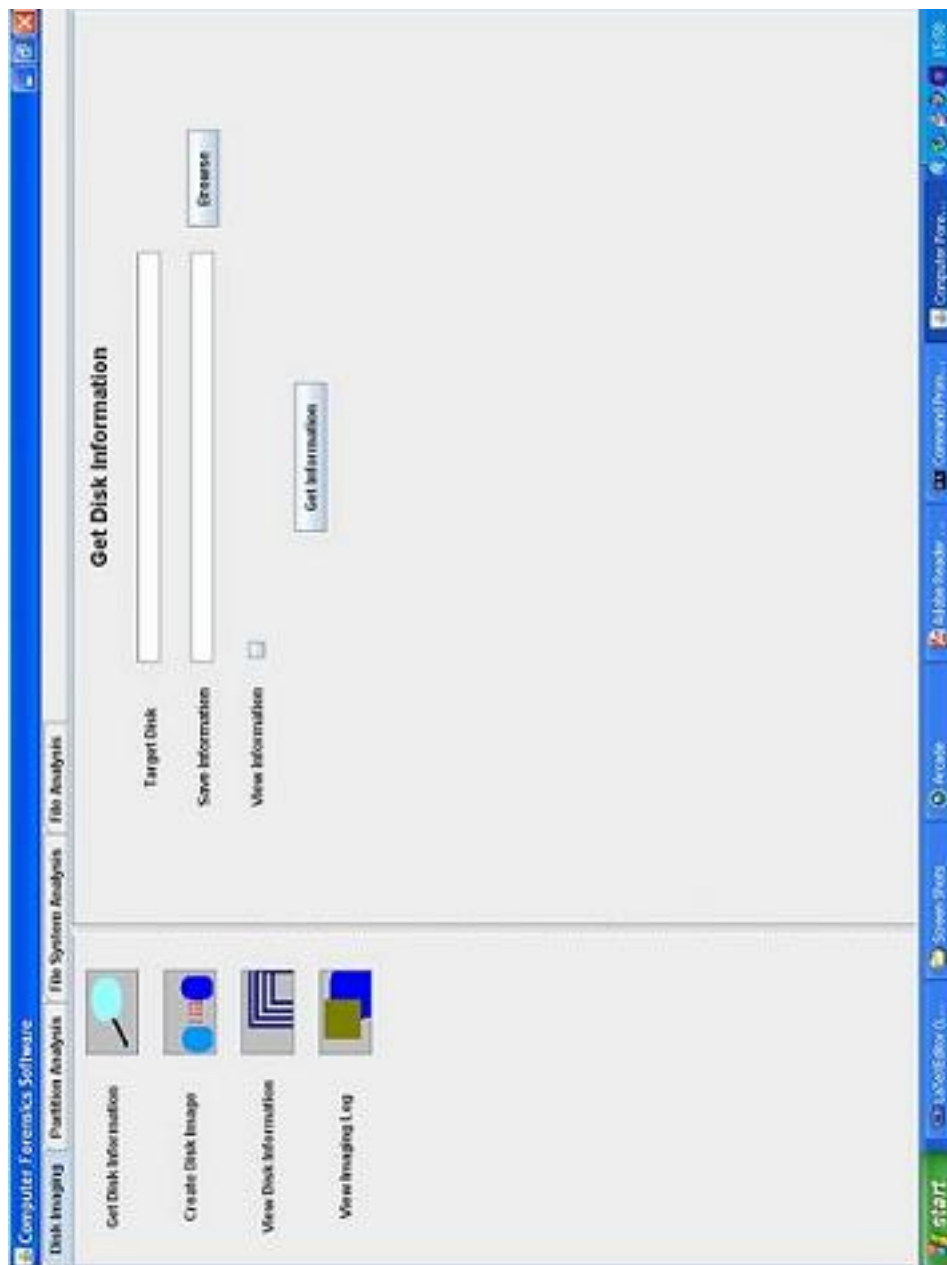


Figure 37: Prototype of the Get Disk Info Menu

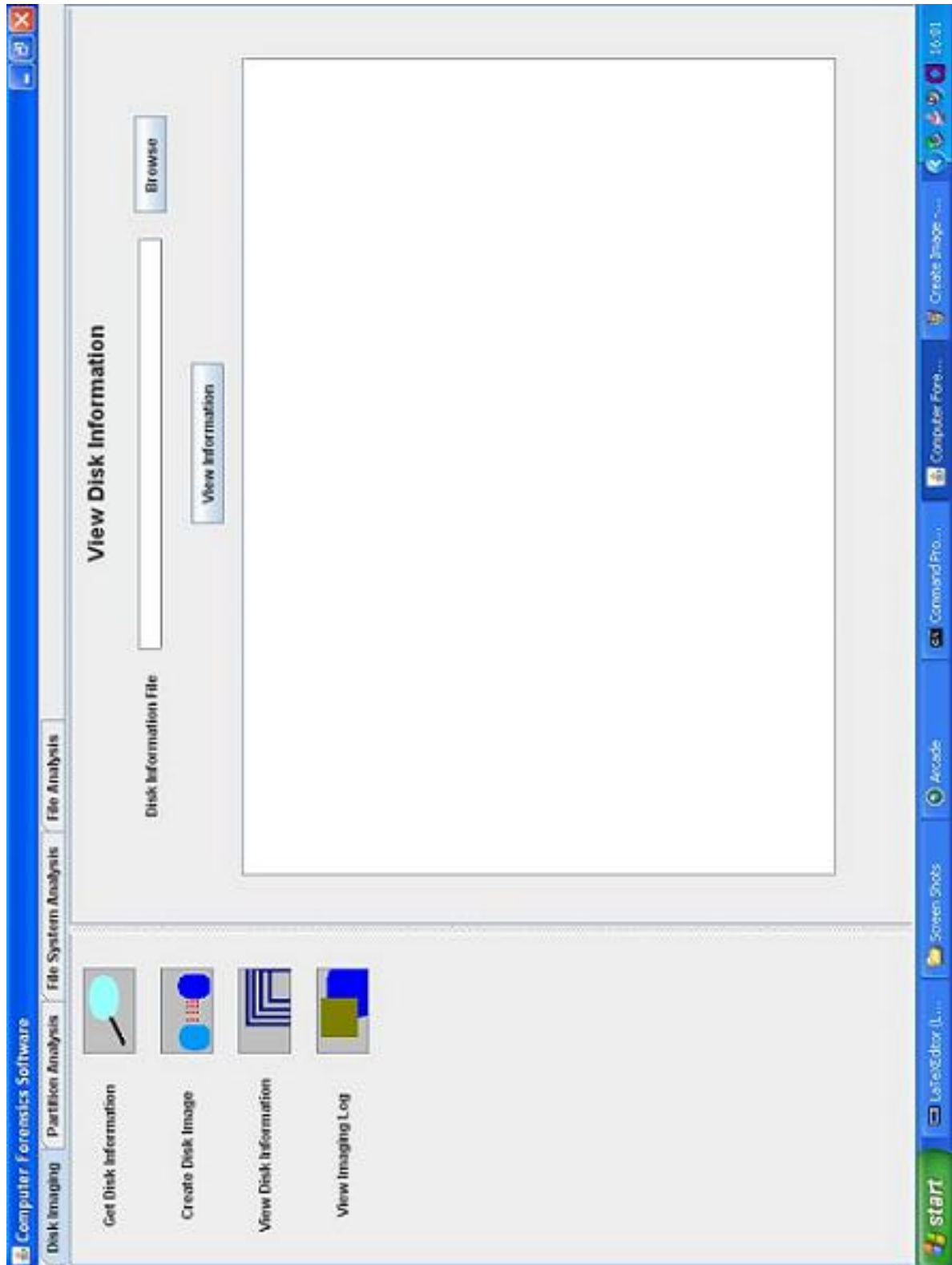


Figure 38: Prototype of the View Disk Info Menu

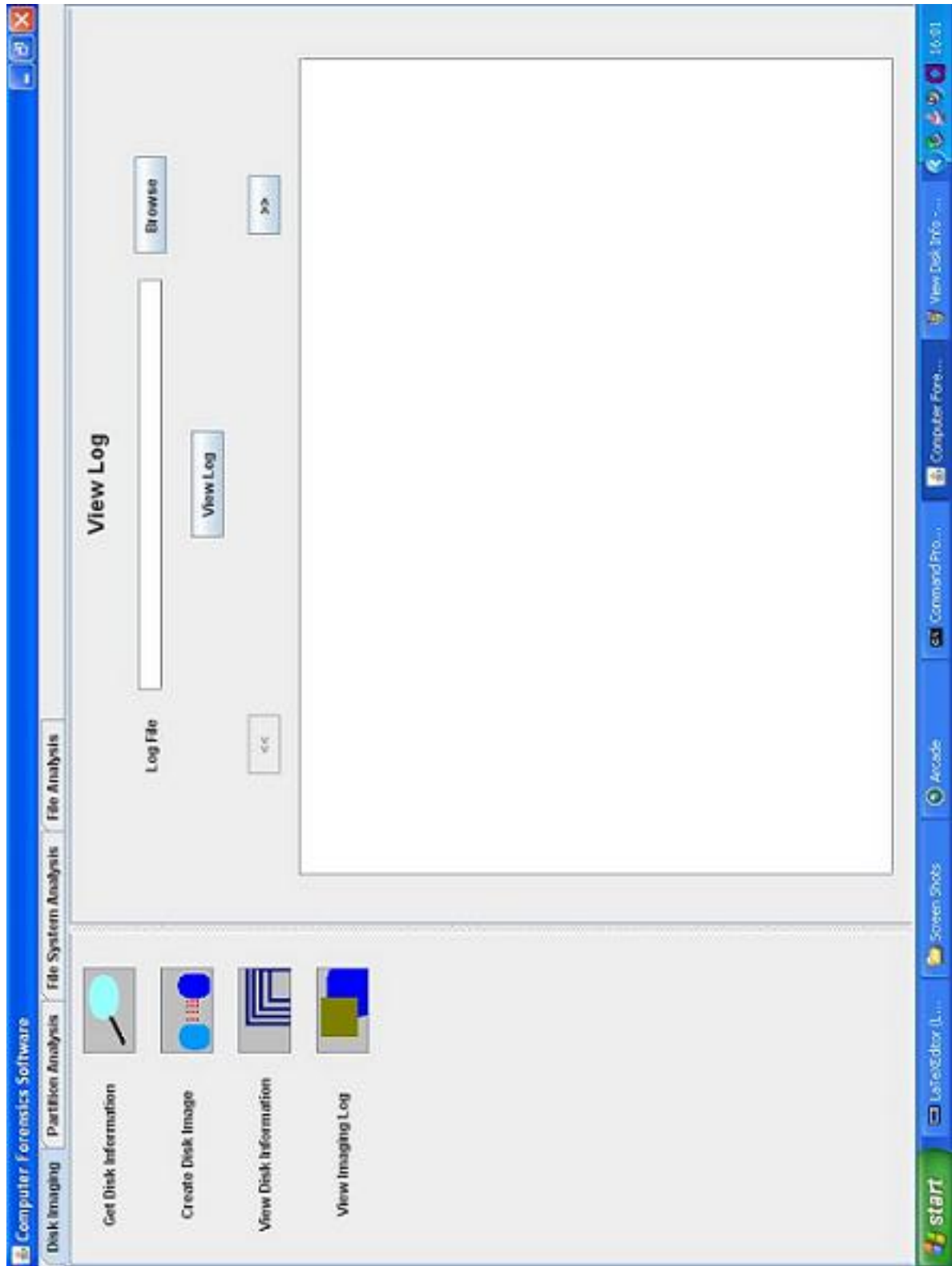


Figure 39: Prototype of the View Imaging Log Menu

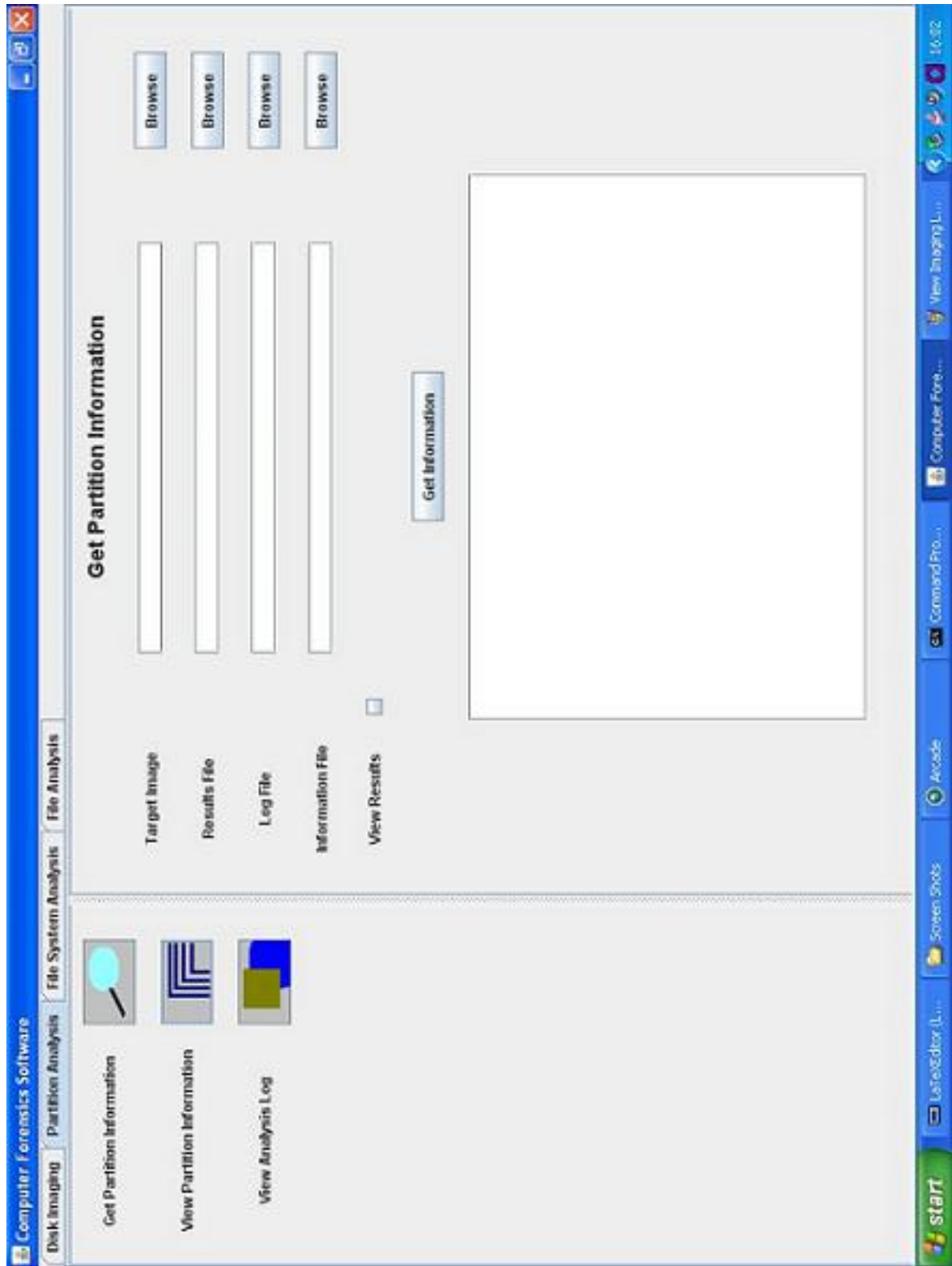


Figure 40: Prototype of the Get Partition Info Menu

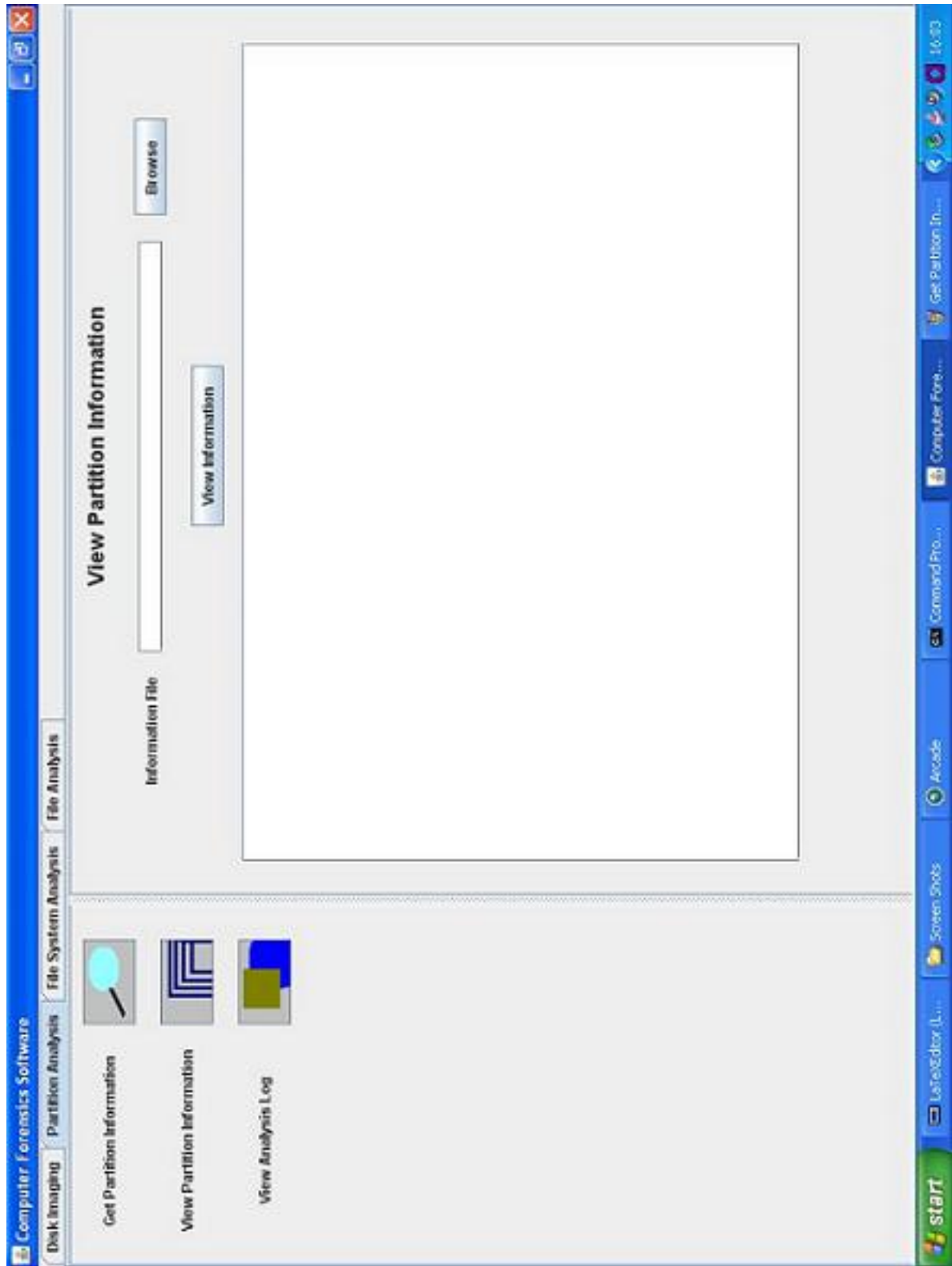


Figure 41: Prototype of the View Partition Info Menu

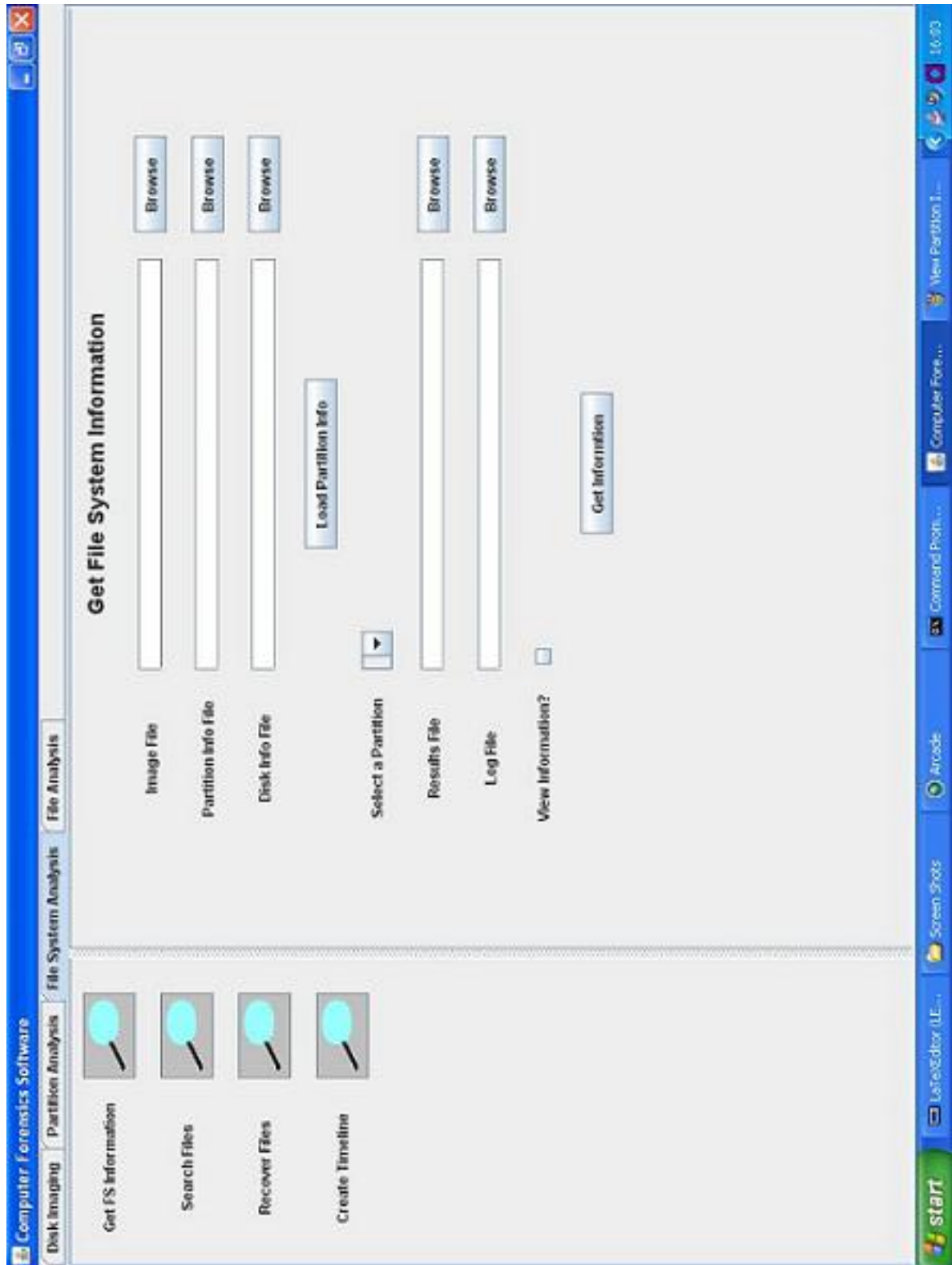


Figure 42: Prototype of the Get File System Info Menu

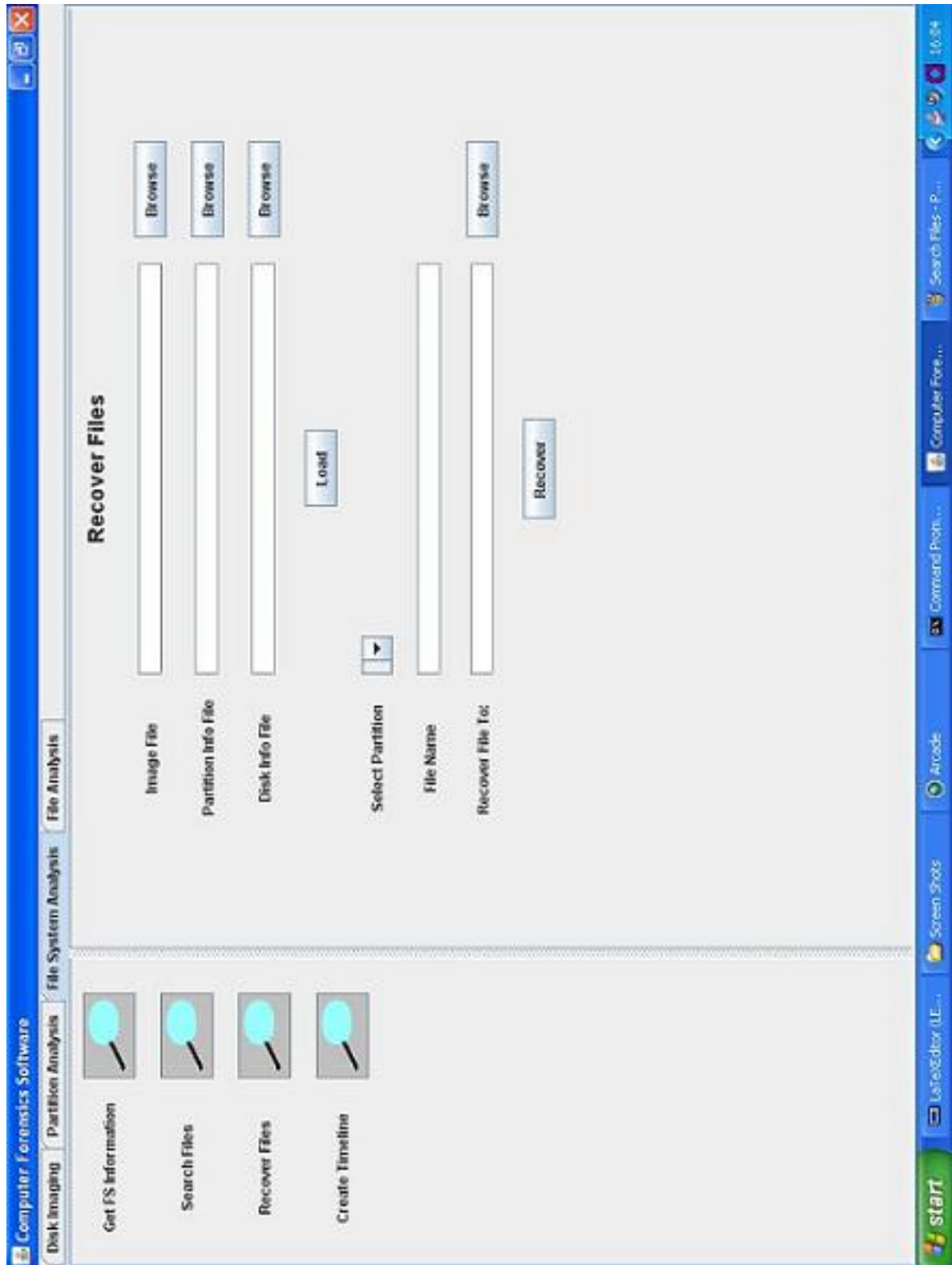


Figure 43: Prototype of the Recover Files Menu

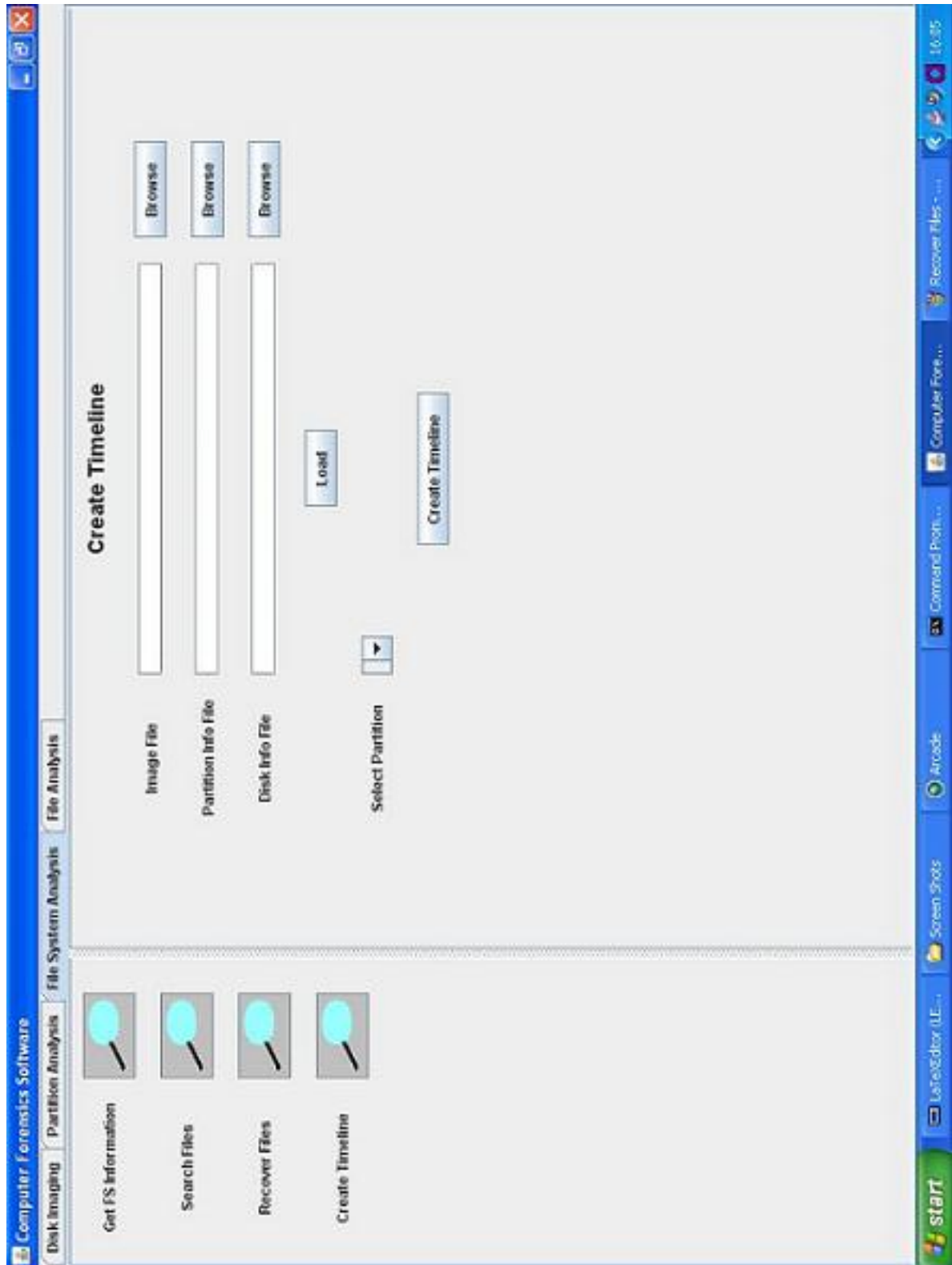


Figure 44: Prototype of Create Timeline Menu

9 Appendix B

9.1 Test Frames

- 15. Information-File-Existence = file doesn't exist
- 16. Information-File-Existence = file exists
Information-File-Validity = file is valid
- 17. Information-File-Existence = file exists
Information-File-Validity = file is not valid
- 18. Log-File-Existence = file doesn't exist
- 19. Log-File-Existence = file exists
Log-File-Validity = file is valid
- 20. Log-File-Existence = file exists
Log-File-Existence = file is not valid
- 21. Image-File-Existence = file doesn't exist
- 22. Results-File-Path-Validity = path is not valid
- 23. Log-File-Path-Validity = path is not valid
- 24. Information-File-Existence = file doesn't exist
- 25. Information-File-Validity = file is not valid
- 26. Image-File-Existence = file exists
Image-File-Validity = file is valid
Results-File-Path-Validity = path is valid
Log-File-Path-Validity = path is valid
Information-File-Existence = file exists
Information-File-Validity = file is valid
Results-To-Be-Displayed = yes
- 27. Image-File-Existence = file exists
Image-File-Validity = file is valid
Results-File-Path-Validity = path is valid
Log-File-Path-Validity = path is valid
Information-File-Existence = file exists
Information-File-Validity = file is valid
Results-To-Be-Displayed = no
- 28. Image-File-Existence = file exists
Image-File-Validity = file is valid
Results-File-Path-Validity = path is not valid
Log-File-Path-Validity = path is valid
Information-File-Existence = file exists
Information-File-Validity = file is valid

Results-To-Be-Displayed = yes

29. Image-File-Existence = file exists
 Image-File-Validity = file is valid
 Results-File-Path-Validity = path is not valid
 Log-File-Path-Validity = path is valid
 Information-File-Existence = file exists
 Information-File-Validity = file is valid
 Results-To-Be-Displayed = no

30. Results-File-Existence = file doesn't exist

31. Results-File-Existence = file exists
 Results-File-Validity = file is valid

32. Results-File-Existence = file exists
 Results-File-Validity = file is not valid

33. Image-File-Existence = file doesn't exist

34. Partition-Information-File-Existence = file doesn't exist

35. Partition-Information-File-Validity = file is not valid

36. Disk-Information-File-Existence = file doesn't exist

37. Disk-Information-File-Validity = file is not valid

38. Image-File-Existence = file exists
 Partition-Information-File-Existence = file exists
 Partition-Information-File-Validity = file is valid
 Disk-Information-File-Existence = file exists
 Disk-Information-File-Validity = file is valid

39. Has-a-Partition-Been-Selected = yes

40. Is-The-Partition-Type-Supported = No

41. Validity-Of-Results-File-Path = path is valid

42. Log-File-Path-Validity = path is not valid

43. Has-a-Partition-Been-Selected = yes
 Is-the-Partition-Type-Supported = no
 Validity-Of-Results-File-Path = path is valid
 Log-File-Path-Validity = path is valid
 Display-Information = Yes

44. Has-a-Partition-Been-Selected = yes
 Is-the-Partition-Type-Supported = no
 Validity-Of-Results-File-Path = path is valid
 Log-File-Path-Validity = path is valid
 Display-Information = No

- 45. Has-A-Partition-Been-Selected = No
- 46. Is-The-Partition-Type-Supported = No
- 47. Has-A-Search-Type-Been-Selected = No
- 48. Has-A-File-Status-Been-Selected = No
- 49. The-Validity-of-the-results-file-path = inavlid path
- 50. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
- 51. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
- 52. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
- 53. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
- 54. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name

- The-File-Types-to-be-Searched = Images
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
55. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Images
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
56. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Images
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
57. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Images
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
58. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
59. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String

- The-Validity-of-the-results-file-path = valid path
60. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Text
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
61. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Text
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
62. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Text
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
63. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Text
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
64. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Text
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
65. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes

- Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
66. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
67. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
68. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
69. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
70. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes

```

The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

71. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

72. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

73. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

74. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

75. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

```

76. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
77. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
78. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
79. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
80. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
81. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name

- The-File-Types-to-be-Searched = Gif
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
82. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
83. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
84. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
85. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
86. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String

- The-Validity-of-the-results-file-path = valid path
87. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
88. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
89. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
90. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
91. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
92. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes

- Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
93. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
94. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
95. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
96. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
97. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes

```

The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

98. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

99. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

100. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

101. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

102. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

```

103. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
104. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
105. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
106. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
107. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
108. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name

The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

109. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

110. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

111. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

112. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

113. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String

The-Validity-of-the-results-file-path = valid path

114. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

115. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

116. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

117. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

118. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Name
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

119. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes

```

Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Rich Text Format
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

120. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Rich Text Format
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

121. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = Rich Text Format
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

122. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

123. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

124. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes

```



```

The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

125. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

126. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

127. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

128. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

129. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Name
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

```

130. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
131. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
132. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
133. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
134. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
135. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content

- The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
136. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
137. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
138. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Text
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
139. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Text
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
140. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Text
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String

The-Validity-of-the-results-file-path = valid path

141. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

142. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

143. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

144. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

145. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

146. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes

```

Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

147. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

148. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

149. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

150. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

151. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes

```

```

The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

152. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

153. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

154. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

155. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

156. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

```

157. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
158. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
159. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
160. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
161. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
162. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content

```

The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

163. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

164. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

165. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

166. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

167. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String

```


The-Validity-of-the-results-file-path = valid path

168. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

169. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

170. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

171. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

172. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

173. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes

```

Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

174. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

175. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

176. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

177. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

178. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes

```

The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

179. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Word Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

180. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Word Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

181. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Word Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

182. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Word Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

183. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Word Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

184. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
185. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
186. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
187. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
188. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
189. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content

- The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
190. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
191. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
192. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path
193. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path
194. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String

The-Validity-of-the-results-file-path = valid path

195. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

196. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

197. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

198. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

199. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = File Content
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

200. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes

```

Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Rich Text Format
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

201. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = Rich Text Format
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

202. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

203. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

204. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

205. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes

```

```

The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

206. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

207. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

208. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

209. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = File Content
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

210. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

```


211. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
212. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
213. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
214. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
215. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Images
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
216. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both

The-File-Types-to-be-Searched = Images
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

217. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Images
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

218. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

219. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

220. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

221. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String

The-Validity-of-the-results-file-path = valid path

222. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

223. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

224. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

225. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

226. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

227. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes

Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

228. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

229. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

230. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

231. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

232. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Jpeg
 Has-A-File-Status-Been-Selected = Yes

```

The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

233. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Jpeg
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

234. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

235. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

236. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

237. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

```

238. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
239. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
240. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
241. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Gif
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
242. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
243. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both

```

The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

244. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

245. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

246. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

247. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

248. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Bitmap
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String

```

The-Validity-of-the-results-file-path = valid path

249. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Bitmap
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

250. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

251. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

252. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

253. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = PNG
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

254. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes


```

Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

255. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

256. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

257. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = PNG
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

258. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

259. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes

```

```

The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

260. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

261. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

262. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

263. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

264. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

```

265. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Word Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
266. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
267. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
268. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path
269. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Text Document
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path
270. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both

The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

271. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

272. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

273. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Text Document
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

274. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

275. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Deleted
 The-Search-Criteria = non-Empty String

The-Validity-of-the-results-file-path = valid path

276. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

277. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Current
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

278. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

279. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = Hidden
 The-Search-Criteria = non-Empty String
 The-Validity-of-the-results-file-path = valid path

280. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes
 Has-A-Search-Type-Selected = Yes
 The-Search-Type-Selected = Both
 The-File-Types-to-be-Searched = Rich Text Format
 Has-A-File-Status-Been-Selected = Yes
 The-File-Status-Selected = All
 The-Search-Criteria = Empty String
 The-Validity-of-the-results-file-path = valid path

281. Has-A-Partition-Been-Selected = Yes
 Is-The-Partition-Type-Supported = Yes

```

Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = Rich Text Format
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

282. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

283. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Deleted
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

284. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

285. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Current
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

286. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes

```

```

The-File-Status-Selected = Hidden
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

287. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = Hidden
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

288. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = Empty String
The-Validity-of-the-results-file-path = valid path

289. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
Has-A-Search-Type-Selected = Yes
The-Search-Type-Selected = Both
The-File-Types-to-be-Searched = All
Has-A-File-Status-Been-Selected = Yes
The-File-Status-Selected = All
The-Search-Criteria = non-Empty String
The-Validity-of-the-results-file-path = valid path

290. Has-A-Partition-Been-Selected = No

291. Is-The-Partition-Type-Supported = No

292. The-Validity-Of-The-Recovered-File-Path = path is invalid

293. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
The-Existence-Of-The-File-To-Recover = file exists
The-Validity-Of-The-Recovered-File-Path = path is valid

294. Has-A-Partition-Been-Selected = Yes
Is-The-Partition-Type-Supported = Yes
The-Existence-Of-The-File-To-Recover = file doesn't exist
The-Validity-Of-The-Recovered-File-Path = path is valid

```

9.2 Test Results

Figure Shows the results of testing the 'Search Files' function.

Test Number	Test Frame	Partition to Analyse	Type Of Search	File Types	File Status	Criteria	Results File	Expected Outcome	Actual Outcome
1	45		File Name	Images	Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
2	46	NTFS (LBA)	File Name	Images	Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
3	47	FAT16	File Name		Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
4	48	FAT16	File Name	Images		"Img1"	results.txt	An error message should be displayed	An error message was displayed
5	49	FAT16	File Name	Images	Deleted	"Img1"	1.txt	An error message should be displayed	An error message was displayed
6	50	FAT16	File Name	Images	Deleted		results.txt	The results should be displayed	The results were displayed
7	51	FAT16	File Name	Images	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
8	52	FAT16	File Name	Images	Current		results.txt	The results should be displayed	The results were displayed
9	53	FAT16	File Name	Images	Current	Img1	results.txt	The results should be displayed	The results were displayed
10	54	FAT16	File Name	Images	Hidden		results.txt	The results should be displayed	The results were displayed
11	55	FAT16	File Name	Images	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
12	56	FAT16	File Name	Images	All		results.txt	The results should be displayed	The results were displayed
13	57	FAT16	File Name	Images	All	Img1	results.txt	The results should be displayed	The results were displayed

14	58	FAT16	File Name	Text	Deleted		results.txt	The results should be displayed	The results were displayed
15	59	FAT16	File Name	Text	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
16	60	FAT16	File Name	Text	Current		results.txt	The results should be displayed	The results were displayed
17	61	FAT16	File Name	Text	Current	Img1	results.txt	The results should be displayed	The results were displayed
18	62	FAT16	File Name	Text	Hidden		results.txt	The results should be displayed	The results were displayed
19	63	FAT16	File Name	Text	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
20	64	FAT16	File Name	Text	All		results.txt	The results should be displayed	The results were displayed
21	65	FAT16	File Name	Text	All	Img1	results.txt	The results should be displayed	The results were displayed
22	66	FAT16	File Name	Jpeg	Deleted		results.txt	The results should be displayed	The results were displayed
23	67	FAT16	File Name	Jpeg	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
24	68	FAT16	File Name	Jpeg	Current		results.txt	The results should be displayed	The results were displayed
25	69	FAT16	File Name	Jpeg	Current	Img1	results.txt	The results should be displayed	The results were displayed
26	70	FAT16	File Name	Jpeg	Hidden		results.txt	The results should be displayed	The results were displayed
27	71	FAT16	File Name	Jpeg	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
28	72	FAT16	File Name	Jpeg lxii	All		results.txt	The results should be displayed	The results were displayed
29	73	FAT16	File Name	Jpeg	All	Img1	results.txt	The results should be displayed	The results were displayed

30	74	FAT16	File Name	Gif	Deleted		results.txt	The results should be displayed	The results were displayed
31	75	FAT16	File Name	Gif	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
32	76	FAT16	File Name	Gif	Current		results.txt	The results should be displayed	The results were displayed
33	77	FAT16	File Name	Gif	Current	Img1	results.txt	The results should be displayed	The results were displayed
34	78	FAT16	File Name	Gif	Hidden		results.txt	The results should be displayed	The results were displayed
35	79	FAT16	File Name	Gif	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
36	80	FAT16	File Name	Gif	All		results.txt	The results should be displayed	The results were displayed
37	81	FAT16	File Name	Gif	All	Img1	results.txt	The results should be displayed	The results were displayed
38	82	FAT16	File Name	Bitmap	Deleted		results.txt	The results should be displayed	The results were displayed
39	83	FAT16	File Name	Bitmap	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
40	84	FAT16	File Name	Bitmap	Current		results.txt	The results should be displayed	The results were displayed
41	85	FAT16	File Name	Bitmap	Current	Img1	results.txt	The results should be displayed	The results were displayed
42	86	FAT16	File Name	Bitmap	Hidden		results.txt	The results should be displayed	The results were displayed
43	87	FAT16	File Name	Bitmap	Hidden	Img1	results.txt	The results should be displayed	The results were displayed

44	88	FAT16	File Name	Bitmap	All		results.txt	The results should be displayed	The results were displayed
45	89	FAT16	File Name	Bitmap	All	Img1	results.txt	The results should be displayed	The results were displayed
46	90	FAT16	File Name	PNG	Deleted		results.txt	The results should be displayed	The results were displayed
47	91	FAT16	File Name	PNG	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
48	92	FAT16	File Name	PNG	Current		results.txt	The results should be displayed	The results were displayed
49	93	FAT16	File Name	PNG	Current	Img1	results.txt	The results should be displayed	The results were displayed
50	94	FAT16	File Name	PNG	Hidden		results.txt	The results should be displayed	The results were displayed
51	95	FAT16	File Name	PNG	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
52	96	FAT16	File Name	PNG	All		results.txt	The results should be displayed	The results were displayed
53	97	FAT16	File Name	PNG	All	Img1	results.txt	The results should be displayed	The results were displayed
54	98	FAT16	File Name	Word Document	Deleted		results.txt	The results should be displayed	The results were displayed
55	99	FAT16	File Name	Word Document	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
56	100	FAT16	File Name	Word Document	Current		results.txt	The results should be displayed	The results were displayed

57	101	FAT16	File Name	Word Document	Current	Img1	results.txt	The results should be displayed	The results were displayed
58	102	FAT16	File Name	Word Document	Hidden		results.txt	The results should be displayed	The results were displayed
59	103	FAT16	File Name	Word Document	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
60	104	FAT16	File Name	Word Document	All		results.txt	The results should be displayed	The results were displayed
61	105	FAT16	File Name	Word Document	All	Img1	results.txt	The results should be displayed	The results were displayed
62	106	FAT16	File Name	Text Document	Deleted		results.txt	The results should be displayed	The results were displayed
63	107	FAT16	File Name	Text Document	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
64	108	FAT16	File Name	Text Document	Current		results.txt	The results should be displayed	The results were displayed
65	109	FAT16	File Name	Text Document	Current	Img1	results.txt	The results should be displayed	The results were displayed
66	110	FAT16	File Name	Text Document	Hidden		results.txt	The results should be displayed	The results were displayed
67	111	FAT16	File Name	Text Document	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
68	112	FAT16	File Name	Text Document	All		results.txt	The results should be displayed	The results were displayed
69	113	FAT16	File Name	Text Document	All	Img1	results.txt	The results should be displayed	The results were displayed
70	114	FAT16	File Name	Rich Text Format	Deleted		results.txt	The results should be displayed	The results were displayed

71	115	FAT16	File Name	Rich Text Format	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
72	116	FAT16	File Name	Rich Text Format	Current		results.txt	The results should be displayed	The results were displayed
73	117	FAT16	File Name	Rich Text Format	Current	Img1	results.txt	The results should be displayed	The results were displayed
74	118	FAT16	File Name	Rich Text Format	Hidden		results.txt	The results should be displayed	The results were displayed
75	119	FAT16	File Name	Rich Text Format	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
76	120	FAT16	File Name	Rich Text Format	All		results.txt	The results should be displayed	The results were displayed
77	121	FAT16	File Name	Rich Text Format	All	Img1	results.txt	The results should be displayed	The results were displayed
78	122	FAT16	File Name	All	Deleted		results.txt	The results should be displayed	The results were displayed
79	123	FAT16	File Name	All	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
80	124	FAT16	File Name	All	Current		results.txt	The results should be displayed	The results were displayed
81	125	FAT16	File Name	All	Current	Img1	results.txt	The results should be displayed	The results were displayed
82	126	FAT16	File Name	All	Hidden		results.txt	The results should be displayed	The results were displayed
83	127	FAT16	File Name	All	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
84	128	FAT16	File Name	All	All		results.txt	The results should be displayed	The results were displayed
85	129	FAT16	File Name	All lxvi	All	Img1	results.txt	The results should be displayed	The results were displayed

86	130		File Content	Images	Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
87	131	NTFS (LBA)	File Content	Images	Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
88	132	FAT16	File Content		Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
89	133	FAT16	File Content	Images		"Img1"	results.txt	An error message should be displayed	An error message was displayed
90	134	FAT16	File Content	Images	Deleted	"Img1"	1.txt	An error message should be displayed	An error message was displayed
91	135	FAT16	File Content	Images	Deleted		results.txt	The results should be displayed	The results were displayed
92	136	FAT16	File Content	Images	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
93	137	FAT16	File Content	Images	Current		results.txt	The results should be displayed	The results were displayed
94	138	FAT16	File Content	Images	Current	Img1	results.txt	The results should be displayed	The results were displayed
95	139	FAT16	File Content	Images	Hidden		results.txt	The results should be displayed	The results were displayed
96	140	FAT16	File Content	Images	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
97	141	FAT16	File Content	Images	All		results.txt	The results should be displayed	The results were displayed
98	142	FAT16	File Content	Images	All	Img1	results.txt	The results should be displayed	The results were displayed

99	143	FAT16	File Content	Text	Deleted		results.txt	The results should be displayed	The results were displayed
100	144	FAT16	File Content	Text	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
101	145	FAT16	File Content	Text	Current		results.txt	The results should be displayed	The results were displayed
102	146	FAT16	File Content	Text	Current	Img1	results.txt	The results should be displayed	The results were displayed
103	147	FAT16	File Content	Text	Hidden		results.txt	The results should be displayed	The results were displayed
104	148	FAT16	File Content	Text	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
105	149	FAT16	File Content	Text	All		results.txt	The results should be displayed	The results were displayed
106	150	FAT16	File Content	Text	All	Img1	results.txt	The results should be displayed	The results were displayed
107	151	FAT16	File Content	Jpeg	Deleted		results.txt	The results should be displayed	The results were displayed
108	152	FAT16	File Content	Jpeg	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
109	153	FAT16	File Content	Jpeg	Current		results.txt	The results should be displayed	The results were displayed
110	154	FAT16	File Content	Jpeg	Current	Img1	results.txt	The results should be displayed	The results were displayed
111	155	FAT16	File Content	Jpeg	Hidden		results.txt	The results should be displayed	The results were displayed
112	156	FAT16	File Content	Jpeg	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
113	157	FAT16	File Content	Jpeg lxviii	All		results.txt	The results should be displayed	The results were displayed
114	158	FAT16	File Content	Jpeg	All	Img1	results.txt	The results should be displayed	The results were displayed

115	159	FAT16	File Content	Gif	Deleted		results.txt	The results should be displayed	The results were displayed
116	160	FAT16	File Content	Gif	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
117	161	FAT16	File Content	Gif	Current		results.txt	The results should be displayed	The results were displayed
118	162	FAT16	File Content	Gif	Current	Img1	results.txt	The results should be displayed	The results were displayed
119	163	FAT16	File Content	Gif	Hidden		results.txt	The results should be displayed	The results were displayed
120	164	FAT16	File Content	Gif	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
121	165	FAT16	File Content	Gif	All		results.txt	The results should be displayed	The results were displayed
122	166	FAT16	File Content	Gif	All	Img1	results.txt	The results should be displayed	The results were displayed
123	167	FAT16	File Content	Bitmap	Deleted		results.txt	The results should be displayed	The results were displayed
124	168	FAT16	File Content	Bitmap	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
125	169	FAT16	File Content	Bitmap	Current		results.txt	The results should be displayed	The results were displayed
125	170	FAT16	File Content	Bitmap	Current	Img1	results.txt	The results should be displayed	The results were displayed
126	171	FAT16	File Content	Bitmap	Hidden		results.txt	The results should be displayed	The results were displayed
127	172	FAT16	File Content	Bitmap	Hidden	Img1	results.txt	The results should be displayed	The results were displayed

128	173	FAT16	File Content	Bitmap	All		results.txt	The results should be displayed	The results were displayed
129	174	FAT16	File Content	Bitmap	All	Img1	results.txt	The results should be displayed	The results were displayed
130	175	FAT16	File Content	PNG	Deleted		results.txt	The results should be displayed	The results were displayed
131	176	FAT16	File Content	PNG	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
132	177	FAT16	File Content	PNG	Current		results.txt	The results should be displayed	The results were displayed
133	178	FAT16	File Content	PNG	Current	Img1	results.txt	The results should be displayed	The results were displayed
134	179	FAT16	File Content	PNG	Hidden		results.txt	The results should be displayed	The results were displayed
135	180	FAT16	File Content	PNG	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
136	181	FAT16	File Content	PNG	All		results.txt	The results should be displayed	The results were displayed
137	182	FAT16	File Content	PNG	All	Img1	results.txt	The results should be displayed	The results were displayed
138	183	FAT16	File Content	Word Document	Deleted		results.txt	The results should be displayed	The results were displayed
139	184	FAT16	File Content	Word Document	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
140	185	FAT16	File Content	Word Document	Current		results.txt	The results should be displayed	The results were displayed

141	186	FAT16	File Content	Word Document	Current	Img1	results.txt	The results should be displayed	The results were displayed
142	187	FAT16	File Content	Word Document	Hidden		results.txt	The results should be displayed	The results were displayed
143	188	FAT16	File Content	Word Document	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
144	189	FAT16	File Content	Word Document	All		results.txt	The results should be displayed	The results were displayed
145	190	FAT16	File Content	Word Document	All	Img1	results.txt	The results should be displayed	The results were displayed
155	191	FAT16	File Content	Text Document	Deleted		results.txt	The results should be displayed	The results were displayed
156	192	FAT16	File Content	Text Document	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
157	193	FAT16	File Content	Text Document	Current		results.txt	The results should be displayed	The results were displayed
158	194	FAT16	File Content	Text Document	Current	Img1	results.txt	The results should be displayed	The results were displayed
159	195	FAT16	File Content	Text Document	Hidden		results.txt	The results should be displayed	The results were displayed
160	196	FAT16	File Content	Text Document	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
161	197	FAT16	File Content	Text Document	All		results.txt	The results should be displayed	The results were displayed
162	198	FAT16	File Content	Text Document	All	Img1	results.txt	The results should be displayed	The results were displayed
163	198	FAT16	File Content	Rich Text Format	Deleted		results.txt	The results should be displayed	The results were displayed

164	199	FAT16	File Content	Rich Text Format	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
165	200	FAT16	File Content	Rich Text Format	Current		results.txt	The results should be displayed	The results were displayed
166	201	FAT16	File Content	Rich Text Format	Current	Img1	results.txt	The results should be displayed	The results were displayed
167	202	FAT16	File Content	Rich Text Format	Hidden		results.txt	The results should be displayed	The results were displayed
168	203	FAT16	File Content	Rich Text Format	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
169	204	FAT16	File Content	Rich Text Format	All		results.txt	The results should be displayed	The results were displayed
170	205	FAT16	File Content	Rich Text Format	All	Img1	results.txt	The results should be displayed	The results were displayed
171	206	FAT16	File Content	All	Deleted		results.txt	The results should be displayed	The results were displayed
172	207	FAT16	File Content	All	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
173	208	FAT16	File Content	All	Current		results.txt	The results should be displayed	The results were displayed
174	209	FAT16	File Content	All	Current	Img1	results.txt	The results should be displayed	The results were displayed
175	210	FAT16	File Content	All	Hidden		results.txt	The results should be displayed	The results were displayed
176	211	FAT16	File Content	All	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
177	212	FAT16	File Content	All	All		results.txt	The results should be displayed	The results were displayed
178	213	FAT16	File Content	All lxxii	All	Img1	results.txt	The results should be displayed	The results were displayed

179	214		Both	Images	Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
180	215	NTFS (LBA)	Both	Images	Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
181	216	FAT16	Both		Deleted	"Img1"	results.txt	An error message should be displayed	An error message was displayed
182	217	FAT16	Both	Images		"Img1"	results.txt	An error message should be displayed	An error message was displayed
183	218	FAT16	Both	Images	Deleted	"Img1"	1.txt	An error message should be displayed	An error message was displayed
184	219	FAT16	Both	Images	Deleted		results.txt	The results should be displayed	The results were displayed
185	220	FAT16	Both	Images	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
186	221	FAT16	Both	Images	Current		results.txt	The results should be displayed	The results were displayed
187	222	FAT16	Both	Images	Current	Img1	results.txt	The results should be displayed	The results were displayed
188	223	FAT16	Both	Images	Hidden		results.txt	The results should be displayed	The results were displayed
189	224	FAT16	Both	Images	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
190	225	FAT16	Both	Images	All		results.txt	The results should be displayed	The results were displayed
191	226	FAT16	Both	Images	All	Img1	results.txt	The results should be displayed	The results were displayed

192	227	FAT16	Both	Text	Deleted		results.txt	The results should be displayed	The results were displayed
193	228	FAT16	Both	Text	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
194	229	FAT16	Both	Text	Current		results.txt	The results should be displayed	The results were displayed
195	230	FAT16	Both	Text	Current	Img1	results.txt	The results should be displayed	The results were displayed
196	231	FAT16	Both	Text	Hidden		results.txt	The results should be displayed	The results were displayed
197	232	FAT16	Both	Text	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
198	233	FAT16	Both	Text	All		results.txt	The results should be displayed	The results were displayed
199	234	FAT16	Both	Text	All	Img1	results.txt	The results should be displayed	The results were displayed
200	235	FAT16	Both	Jpeg	Deleted		results.txt	The results should be displayed	The results were displayed
201	236	FAT16	Both	Jpeg	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
202	237	FAT16	Both	Jpeg	Current		results.txt	The results should be displayed	The results were displayed
203	238	FAT16	Both	Jpeg	Current	Img1	results.txt	The results should be displayed	The results were displayed
204	239	FAT16	Both	Jpeg	Hidden		results.txt	The results should be displayed	The results were displayed
205	240	FAT16	Both	Jpeg	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
206	241	FAT16	Both	Jpeg lxxiv	All		results.txt	The results should be displayed	The results were displayed
207	242	FAT16	Both	Jpeg	All	Img1	results.txt	The results should be displayed	The results were displayed

208	243	FAT16	Both	Gif	Deleted		results.txt	The results should be displayed	The results were displayed
209	244	FAT16	Both	Gif	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
210	245	FAT16	Both	Gif	Current		results.txt	The results should be displayed	The results were displayed
211	246	FAT16	Both	Gif	Current	Img1	results.txt	The results should be displayed	The results were displayed
212	247	FAT16	Both	Gif	Hidden		results.txt	The results should be displayed	The results were displayed
213	248	FAT16	Both	Gif	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
214	249	FAT16	Both	Gif	All		results.txt	The results should be displayed	The results were displayed
215	250	FAT16	Both	Gif	All	Img1	results.txt	The results should be displayed	The results were displayed
216	251	FAT16	Both	Bitmap	Deleted		results.txt	The results should be displayed	The results were displayed
217	252	FAT16	Both	Bitmap	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
218	253	FAT16	Both	Bitmap	Current		results.txt	The results should be displayed	The results were displayed
219	254	FAT16	Both	Bitmap	Current	Img1	results.txt	The results should be displayed	The results were displayed
220	255	FAT16	Both	Bitmap	Hidden		results.txt	The results should be displayed	The results were displayed
221	256	FAT16	Both	Bitmap	Hidden	Img1	results.txt	The results should be displayed	The results were displayed

222	257	FAT16	Both	Bitmap	All		results.txt	The results should be displayed	The results were displayed
223	258	FAT16	Both	Bitmap	All	Img1	results.txt	The results should be displayed	The results were displayed
224	259	FAT16	Both	PNG	Deleted		results.txt	The results should be displayed	The results were displayed
225	260	FAT16	Both	PNG	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
226	261	FAT16	Both	PNG	Current		results.txt	The results should be displayed	The results were displayed
227	262	FAT16	Both	PNG	Current	Img1	results.txt	The results should be displayed	The results were displayed
228	263	FAT16	Both	PNG	Hidden		results.txt	The results should be displayed	The results were displayed
229	264	FAT16	Both	PNG	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
230	265	FAT16	Both	PNG	All		results.txt	The results should be displayed	The results were displayed
231	266	FAT16	Both	PNG	All	Img1	results.txt	The results should be displayed	The results were displayed
232	267	FAT16	Both	Word Document	Deleted		results.txt	The results should be displayed	The results were displayed
233	268	FAT16	Both	Word Document	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
234	269	FAT16	Both	Word Document	Current		results.txt	The results should be displayed	The results were displayed

235	270	FAT16	Both	Word Document	Current	Img1	results.txt	The results should be displayed	The results were displayed
236	271	FAT16	Both	Word Document	Hidden		results.txt	The results should be displayed	The results were displayed
237	272	FAT16	Both	Word Document	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
238	273	FAT16	Both	Word Document	All		results.txt	The results should be displayed	The results were displayed
239	274	FAT16	Both	Word Document	All	Img1	results.txt	The results should be displayed	The results were displayed
240	275	FAT16	Both	Text Document	Deleted		results.txt	The results should be displayed	The results were displayed
241	276	FAT16	Both	Text Document	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
242	277	FAT16	Both	Text Document	Current		results.txt	The results should be displayed	The results were displayed
243	278	FAT16	Both	Text Document	Current	Img1	results.txt	The results should be displayed	The results were displayed
244	279	FAT16	Both	Text Document	Hidden		results.txt	The results should be displayed	The results were displayed
244	280	FAT16	Both	Text Document	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
245	281	FAT16	Both	Text Document	All		results.txt	The results should be displayed	The results were displayed
246	282	FAT16	Both	Text Document	All	Img1	results.txt	The results should be displayed	The results were displayed
247	283	FAT16	Both	Rich Text Format	Deleted		results.txt	The results should be displayed	The results were displayed

248	284	FAT16	Both	Rich Text Format	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
249	285	FAT16	Both	Rich Text Format	Current		results.txt	The results should be displayed	The results were displayed
250	286	FAT16	Both	Rich Text Format	Current	Img1	results.txt	The results should be displayed	The results were displayed
251	287	FAT16	Both	Rich Text Format	Hidden		results.txt	The results should be displayed	The results were displayed
252	288	FAT16	Both	Rich Text Format	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
253	289	FAT16	Both	Rich Text Format	All		results.txt	The results should be displayed	The results were displayed
254	290	FAT16	Both	Rich Text Format	All	Img1	results.txt	The results should be displayed	The results were displayed
255	291	FAT16	Both	All	Deleted		results.txt	The results should be displayed	The results were displayed
256	292	FAT16	Both	All	Deleted	Img1	results.txt	The results should be displayed	The results were displayed
257	293	FAT16	Both	All	Current		results.txt	The results should be displayed	The results were displayed
258	294	FAT16	Both	All	Current	Img1	results.txt	The results should be displayed	The results were displayed
259	295	FAT16	Both	All	Hidden		results.txt	The results should be displayed	The results were displayed
260	296	FAT16	Both	All	Hidden	Img1	results.txt	The results should be displayed	The results were displayed
261	297	FAT16	Both	All	All		results.txt	The results should be displayed	The results were displayed
262	298	FAT16	Both	All lxxviii	All	Img1	results.txt	The results should be displayed	The results were displayed