DIAMOND EDGE

VERSION 2

Diagnostics

Disk Repair

Optimize

Recovery

Innovative Software from

OREGON RESEARCH

DIAMOND EDGE

Disk Diagnostics and Repair for your Atari

Reference Manual

OREGON RESEARCH

Oregon Research
16200 S.W. Pacific Hwy., Suite 162
Tigard, OR 97224

Customer Support: (503) 620-4919
FAX: (503) 624-2940
Internet: orres@teleport.com
Genie: ORA
CompuServe: 71333,2655

Diamond Edge Program Design: Bob Luneski
Diamond Edge Programming: Bob Luneski
DADE Program Design: Erin Monaco and Bob Luneski
DADE Programming: Erin Monaco
DADE Physical Disk Driver: Bob Luneski
Manual by: Bob Luneski

My everloving thanks goes to
my wife Jo,
son David,
and daughter Sara.

My strength and inspiration are drawn from the family core. Without whose constant love and support this project would never have been completed.

Software License Agreement

THE INSTALLATION OF "DIAMOND EDGE", HEREINAFTER REFERRED TO AS "THE SOFTWARE", INDICATES YOUR EXPLICIT AGREEMENT TO THE TERMS AND CONDITIONS OF THIS SOFTWARE LICENSE AGREEMENT.

OREGON RESEARCH GRANTS YOU A NONEXCLUSIVE LICENSE TO INSTALL AND USE THE SOFTWARE ON A SINGLE MACHINE DESIGNATED BY YOU. BACKUP COPIES OF THE SOFTWARE MAY BE MADE FOR ARCHIVAL PURPOSES ONLY. RENTAL OF THIS SOFTWARE IS EXPRESSLY PROHIBITED BY THIS LICENSE.

OREGON RESEARCH MAKES NO WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. REGARDING THE SOFTWARE, OREGON RESEARCH DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE SOFTWARE IN TERMS OF ITS CORRECTNESS, ACCURACY, RELIABILITY, CURRENTNESS, OR OTHERWISE. THE ENTIRE RISK AS TO THE RESULTS AND THE PERFORMANCE OF THE SOFTWARE IS ASSUMED BY YOU. THE EXCLUSION OF IMPLIED WARRANTIES IS NOT PERMITTED BY SOME STATES. THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.

IN NO EVENT WILL OREGON RESEARCH OR IT'S AGENTS BE LIABLE TO YOU FOR ANY CONSEQUENTIAL, INCIDENTAL, OR INDIRECT DAMAGES(INCLUDING DAMAGES FOR THE LOSS OF BUSINESS PROFITS, LOSS OF BUSINESS INFORMATION AND THE LIKE) ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE EVEN IF OREGON RESEARCH HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.

OREGON RESEARCH'S LIABILITY TO YOU FOR ACTUAL DAMAGES FROM ANY CAUSE WHATSOEVER, AND REGARDLESS OF THE FORM OF THE ACTION (WHETHER IN CONTRACT, TORT(INCLUDING NEGLIGENCE), PRODUCT LIABILITY OR OTHERWISE), WILL BE LIMITED TO $50.

Table of Contents

1. Getting Started

What can Diamond Edge can do for you?

Congratulations, you have just purchased the most powerful set of disk management, optimization, diagnostic, repair, and data recovery tools available for the Atari ST. One of the most frequently used and least appreciated components of your computer system are your disks and disk drives. That is, until something goes wrong. Diamond Edge provides you with easy to use disk management, optimization, diagnostic, repair, and recovery tools. With Diamond Edge you can:

- Optimize your hard drive for reading or writing speed
- Save, restore, and edit critical disk information to repair damaged information or recover crashed Hard disks.
- Validate the files on your hard disk with checksums or CRCs to guarantee accuracy and detect file corruption.
- Test and repair the structure of your disks. Diamond Edge can identify and repair an exhaustive range of disk problems and can recover information even from very badly damaged disks.
- Map bad sectors on your hard disk to identify physically damaged portions of your disks and mark them unavailable for use. It will even repair and recover information from any effected files.
- Recover deleted or damaged files
- Obtain critical information about your disk status
- Partition your hard disk
- Zero, wipe, and unzero partitions
- Copy partitions either as an image or defragmenting while copying
- Perform all of these functions on hard disks or floppy disks
- Includes Diamond Mirror to test and save critical disk data on bootup. This makes possible the recovery of even fragmented deleted files.
- Includes Diamond Advanced Disk Editor(DADE) to allow you to edit physical and logical hard disk information as well as binary information within files.
- DADE allows you to edit disks at the Physical device level. There is no disk problem that you can't fix with this tool.

Installation

Included with your software distribution is an owner's registration card. Please take the time to fill it out completely and return it to Oregon Research **TODAY!** It is very important that we be able to reach you to inform you of product upgrades and other Oregon Research news. Before you may begin to use you new software you must complete the installation procedure. Insert your Diamond Edge distribution disk into your floppy drive and Double-click on INSTALL.PRG.

To complete the installation you must provide your name, address, product serial number(located on your distribution disk), and the hard disk partition where you want to install Diamond Edge. You also have the option of installing Diamond Disk Mirror(highly recommended).

When you have completed providing the required information, click on **Install** begin the installation process. The installation program will then create a directory called EDGE on the specified drive and copy all of the required files to your hard drive. If you chose to install Diamond Disk Mirror at this time, then it will be installed in your AUTO folder on Drive C:. Otherwise, it will be copied to your EDGE folder.

You are now ready to enjoy your new software. To begin using Diamond Edge, open the EDGE folder on your hard drive and double click on **EDGE.PRG** to launch the program. Included with your software distribution is an owner's registration card. Please take the time to fill it out completely and return it to Oregon Research **TODAY!** It is very important that we be able to reach you to inform you of product upgrades and other Oregon Research news. A completed registration card is our only way to know how to contact you. It is also a **requirement** to obtain product support services.

What you need to know and do first

Diamond Edge is a very powerful program. However, if used improperly, it can cause as much damage as it can do good.

IT IS IMPERATIVE THAT YOU READ THIS MANUAL COMPLETELY AND CAREFULLY TO FULLY UNDERSTAND THE OPERATION OF DIAMOND EDGE.

Items within the manual that you should pay special attention to are preceded by a ✔.

✔ If you do not understand an operation or the consequences of any program function, then take the time to refer to the manual for clarification. Think before you click!

Here are a few initial actions that you should take after you install Diamond Edge. This will create an emergency disk that can help you recover from many catastrophic disk situations.

✔ **CAREFULLY READ THIS ENTIRE MANUAL**

- After installation, make a backup copy of your master disk and label it **DIAMOND EDGE EMERGENCY.**
- Run Diamond Edge from the Emergency floppy disk.
- Under the Archive Menu, select Save SCSI Information. Select all of your SCSI devices and then select Save SCSI Information.
- Under the Archive Menu, select Save Disk Information. Select all of your partitions and then select Save Disk Information.
- Quit the Program and remove your emergency disk from the floppy drive. Enable the write protect tab on your emergency disk and then store it in a safe place. You should update the disk information weekly and SCSI information whenever your partitions change.
- Backup your hard drive - FREQUENTLY! Diamond Edge will assist you in recovering from many catastrophic disk situations and regain speed through optimization, however there is NO substitute for a valid up to date backup. To make this job as pleasant and fast as possible, we suggest that you use Diamond Back 3 to fulfill your backup requirements.

✔ **CAREFULLY READ THIS MANUAL AGAIN!**

✔ Think before you click! When in doubt, please reread the appropriate manual section. Remember that if improperly used, the tools in this program can cause irrecoverable data loss. Always back up your disk before performing any diagnostic or repair function.

2. The Anatomy of a Disk

The disk drives in your Atari ST are as amazing and mysterious as any part of your computer. The following is intended as a basic primer to Atari ST disk structure. It is does not represent an exhaustive or totally comprehensive treatment of Atari Disk Drives. It is intended solely to provide enough of an understanding of your disks to get full utilization of the power of Diamond Edge.

A basic level of computer knowledge is required to fully understand this section. The user should be familiar with basic storage elements such as bits, bytes, and hexadecimal representation of numbers(denoted by a leading 0x). If the reader is not familiar with these concepts, reference to any good introductory programming book is recommended if you have difficulty understanding this section.

Disk Media and Interfaces

There are two basic types of disks that the Atari ST can access: Floppy disks and Hard Disks. All of Diamond Edges functions are usable on either type of disk. Floppy disks for the Atari ST come in 360K, 720K and 1440K capacities. High density floppies(1440K) require a high density disk and controller upgrade from Atari Corp. or a third party vendor. In addition to greater storage capacity, high density disk drives provide twice the data transfer rate. We highly recommend upgrading your disks to high density compatibility.

Hard disks are the primary storage system for most Atari users. Unlike Floppy disks, hard disks are rigid platters that rotate over fixed heads. The read/write heads never actually come in contact with the disk media, but ride over a extremely small cushion of air. If the heads ever did come in contact with the media, for instance as a result of bumping your hard drive, you would experience significant loss of data. This is known as a head crash and can ruin your hard drive.

The Atari ST communicates with the hard drive via an interface protocol. Until the Atari TT, all Atari ST machines communicated with external devices through a protocol known as the Atari Computer Systems Interface(ACSI). This is similar to, but not identical to the industry standard peripheral communication protocol called the Small Computer Systems Inferface(SCSI).

Most hard drive manufacturers produce hard drives that communicate with the host computer using the standard SCSI protocol. Because the ACSI protocol is not exactly like the SCSI protocol, Atari ST systems require a communication translator to convert the native ACSI protocol to the standard SCSI protocol that the hard drive understands. These are commonly referred to as host adapters. Atari Corp., Supra Corp., ICD Inc., and Berkeley Microsystems all manufacture ACSI host adapters for the Atari ST. In addition to the ACSI interface, the Atari TT and the Falcon 030 also have true SCSI interfaces.

Some older hard drive systems were built using IBM PC compatible MFM or RLL hard drives. These drives not only require the ACSI host adapter to translate ACSI to SCSI, but also require an additional communication translator to convert SCSI commands to MFM/RLL commands. The two translations of commands combined with the general poor performance of MFM drives make these systems extremely slow.

Physical Sector 0

The basic unit of storage on a hard disk is the physical block or sector. The standard size of physical sectors is 512 bytes. The most important sector of all is physical sector 0. This is where the information is stored that tells the hard disk driver how the drive is partitioned and how to access the information stored on the drive. The hard disk driver software reads this sector to determine the logical layout of the physical disk.

The Partition Structure

The partitioning information appears in physical sector zero in the following format. All offsets given are in hexadecimal values from the start of the sector.

Physical Sector 0 Structure

Component	Offset
Total Hard Disk Size	0x1C2
Partition 1 Structure	0x1C6
Partition 2 Structure	0x1D2
Partition 3 Structure	0x1DE
Partition 4 Structure	0x1EA
Start of Bad Sector List	0x1F6
Size of Bad Sector List	0x1FE

Each partition structure contains the following information:

Component	Size
Partition Flag	1 byte
Partition Identifier	3 bytes
Partition Starting Sector	4 bytes
Partition Size	4 bytes

The partition flag defines whether the partition is active and whether the partition is bootable. When set, bit 0 indicates the partition is active and bit 7 indicates that the partition is bootable. Bits 1-6 are currently not used.

The partition identifier is a three character identifier that is used to classify the type of partition. There are three standard types of partitions recognized: GEM, BGM, and XGM.

A GEM partition is the normal type of disk partition. The logical sector size is the same as the physical sector size: 512 bytes. GEM partitions can be as large as 16 megabytes under TOS 1.0 and 1.2 or 32 megabytes(with the right driver) under TOS1.4 or above.

BGM stands for Big GEM partition. These partitions are used to create partitions larger than 16 megabytes. The ST operating system has a limitation on the number of allocation blocks that can be accessed on a single disk. To get around that limitation, and to create larger partitions, the logical size of these allocation units is increased. One logical sector can contain many physical sectors. The logical sector size is a power of 2 times 512 bytes, i.e. logical sectors can be 512, 1024, 2048, 4096, etc. bytes large. The hard disk driver software then converts the request for a logical sector to the appropriate physical sector.

XGM stands for Extended GEM partition. The original Atari hard disk specification as shown above was limited to 4 partitions per hard disk. To get around that limitation, Atari created the specification for the extended partition. The XGM entry points to a location on the hard disk where another psuedo-physical sector 0 resides that describes the location and size of additional partitions. These additional psuedo-root sectors can be chained together to create as many partitions as are required.

Other partition types have been defined by a variety of different operating environments outside of TOS. ACK and OOP are used as identifiers for Spectre partitions. OS9 is used as the partition identifier for the OS9 operating system.

The remaining entries in the partition structure pertain to the partition size and location. The starting sector is the starting physical sector of the partition. The partition boot sector always resides in the first logical sector of the partition. The sector size is the total number of 512 byte physical sectors in the partition.

Vendor Specific Considerations

Several third party hard disk host adapter vendors developed an extended partitioning standard prior to the release of Atari's standard. In particular, Supra Corporation and ICD hard drives support this standard. The basic difference is that these vendors allow up to 12 individual partition structures to be specified in physical sector 0 before an extended partition, described above, is required. The format of the partition information structure is identical, they just allow more to be specified in each sector. The offsets of these additional partition structures are as follows:

Third Party Partition Extensions

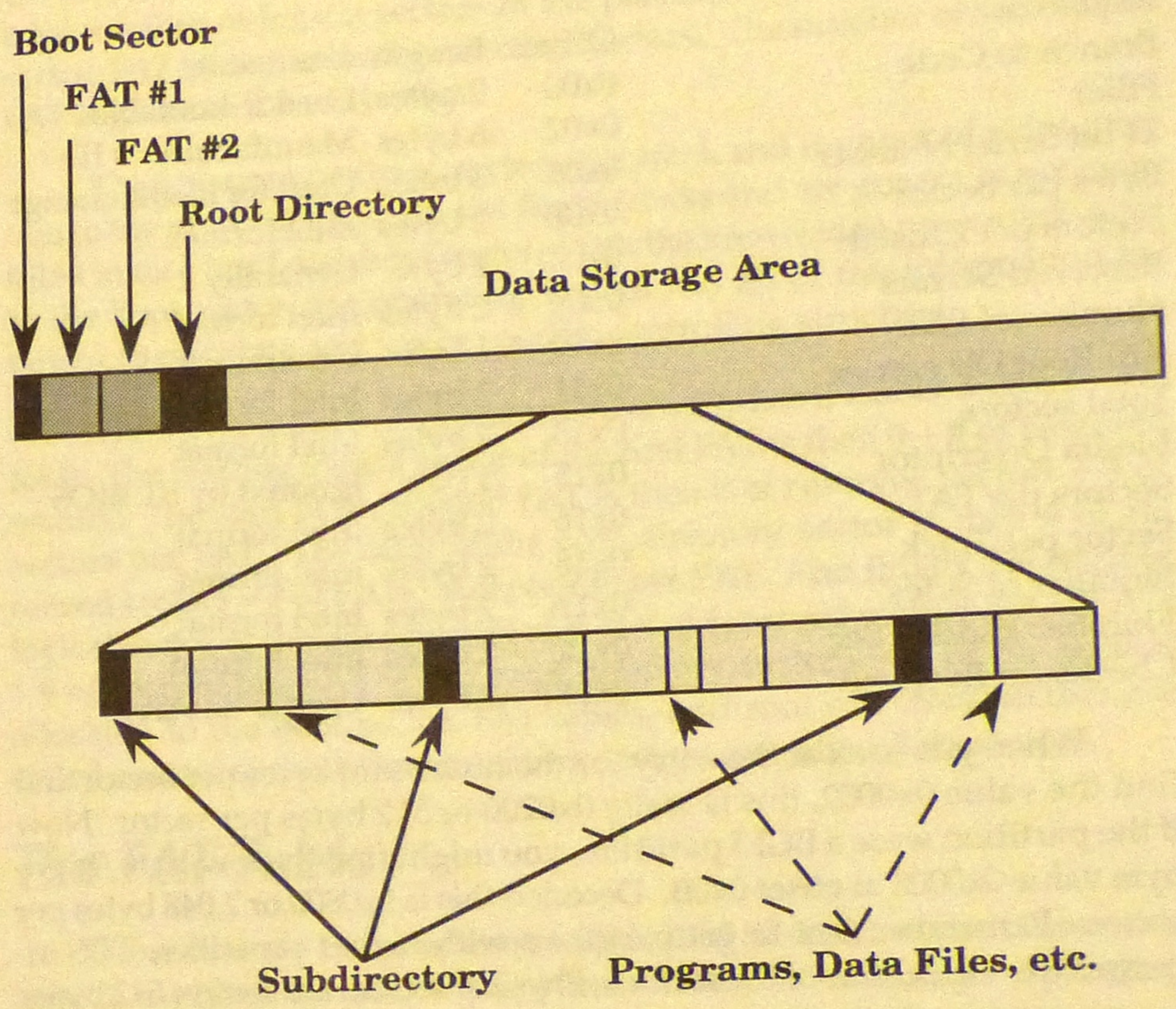
Component	Offset
Partition 5 Structure	0x156
Partition 6 Structure	0x162
Partition 7 Structure	0x16E
Partition 8 Structure	0x17A
Partition 9 Structure	0x186
Partition 10 Structure	0x192
Partition 11 Structure	0x19E
Partition 12 Structure	0x1AA

It is important to note that an ICD or Supra hard disk driver can access all of the partitions on a Atari formatted hard disk. However, the reverse is not true. An Atari hard disk driver can only access the first four partitions on a ICD or Supra formatted drive.

Inside a Disk Partition

The internal structure of a disk drive partition is really not as complicated as it seems. It is organized via a controlling boot sector that tells the hard disk driver how big the drive is and where to find key disk information. Information about files and file attributes are stored in directory sectors. The information about where files are physically located on the disk is contained in the File Allocation Table or FAT. A linear layout of your disk is shown on the next page.

Disk Structure



The Boot Sector

The partition boot sector is one of the most important sectors on each partition. The boot sector always occupies logical sector 0 of the partition. It tells the hard disk driver all of the critical disk parameters necessary to access information on the disk. Most of the Diamond Edge Disk Information screen is based on information contained in the boot sector. The information contained in the boot sector and each items significance is described below. All offsets given are in hexadecimal values from the start of the sector.

Most of the information stored in the boot sector is coded in 8086 or Intel format. This means the first byte represents the low byte of the word and the second byte represents the high byte of the word. This is opposite of a Motorola stored word which is stored high byte first and low byte second. The reason that the ST uses the Intel encoding in the boot sector is because the Atari ST disk structure was derived from and is compatible with IBM disk structure.

Boot Sector Contents

Item	Offset	Length	Comments
Branch to Code	0x00	2 bytes	Used if bootable
Filler	0x02	6 bytes	Manufacturer's ID
24 bit Serial Number	0x08	3 bytes	Used for media change
Bytes per Sector	0x0B	2 bytes	Intel format
Sectors per Cluster	0x0D	1 byte	Generally 2
Reserved Sectors	0x0E	2 bytes	Intel format
Number of FATs	0x10	1 byte	Generally 2
# of Root Dir entries	0x11	2 bytes	Intel format
Total sectors	0x13	2 bytes	Intel format
Media Descriptor	0x15	1 byte	Ignored by ST BIOS
Sectors per FAT	0x16	2 bytes	Intel format
Sector per Track	0x18	2 bytes	Intel format
Number of Sides	0x1A	2 bytes	Intel format
Number Hidden Sec	0x1C	2 bytes	Intel Format
Checksum Even out	0x1FE	2 bytes	Used if Bootable

When you look at the entry for the number of bytes per sector and find the value 0x0002, this is really 0x0200 or 512 bytes per sector. Now if the partition were a BGM partition, you might find the low-byte/high-byte value 0x0008 at offset 0x0B. Decoded this is 0x0800 or 2,048 bytes per sector. Remember that to get partitions with higher capacities, TOS increases the logical size of the sectors. Physical sectors are always 512 bytes.

The next item in the boot sector is the number of sectors per cluster. Typically, there are 2 sectors per cluster. A cluster is the basic disk allocation unit. All space allocated to files is allocated in whole numbers of clusters and the smallest number of clusters allocated to a non-zero length file is one. For standard GEM partitions with 512 byte logical sector sizes, this means that a small file will consume 1 cluster = 2 sectors = 1,024 bytes of disk capacity regardless if the file is 1 byte large or 1,024 bytes large. One of the disadvantages of BGM partitions is inefficient disk utilization. The same 1 byte file that occupied 1,024 bytes on a GEM partition, would occupy 4,096 bytes on a BGM partition with 2,048 byte sectors.

The number of reserved sectors is the number of sectors reserved at the start of the partition including the boot sector. This value is typically equal to one. The number of File Allocation Tables is typically equal to 2. These structures will be described in a later section.

The next entry is the maximum number of root directory entries. Unlike subdirectories, the root directory is a fixed size. Since each directory entry occupies 32 bytes, the number of logical sectors occupied by

the root sector can be calculated from this value. The next entry is the total number of logical sectors in the partition. This includes the boot sector, FAT tables, and root directory sectors. The number of sectors per FAT is given in the next entry.

The remaining entries, sectors per track and number of sides, pertain to the physical properties of floppy disks and are generally not even filled in on a hard disk boot sector. With the knowledge presented so far in the boot sector, the operating system has all of the information required to calculate and locate specific controlling structures.

The sector numbering convention always has the boot sector as sector 0. The starting logical sector of the first FAT is then 0 + # of reserved sectors. The start of the second FAT is then # of reserved sectors + # of sectors per FAT. The start of the root directory sector is then # of reserved sectors + # of FATs * # of sectors per FAT. And finally the starting logical sector of the data storage area is # of reserved sectors + # of FATs * # of sectors per FAT + # of root directory sectors. All of the sectors allocated to the boot sector, FAT tables, and root directory sectors are called control sectors, the remaining sectors are called data sectors.

The FAT Table

Now that we know where to find all of the control structures, it is time to examine the most critical one. In fact so critical, the operating system maintains two independent copies of the FAT table to guard against corruption and unrecoverable loss of data. The File Allocation Table is used to store information about where files reside on the disk, what space is used by files and what space is available for use. Simply stated, the File Allocation Table is a road map to tell the operating system where to go look for an existing file and where it can put new ones.

The values of each entry in the FAT are encoded. The Atari ST supports two styles of encoding for the FAT: 12 bit encoding and 16 bit encoding. 12 bit encoding is generally used for small capacity(no more than 4096 clusters) storage media such as floppy disks and RAM disks because it reduces the size required for the FAT. In this encoding scheme, two FAT entries occupy three bytes, whereas in 16 bit encoding two FAT entries occupy 4 bytes. Decoding a 12 bit FAT is quite complicated, however 16 bit FAT encoding is simply the Intel encoding low byte-high byte described earlier.

The first cluster available for data storage is cluster number 2. For each file, the value of each entry in the FAT is the next cluster containing data for that file.

There are several special values that a FAT entry can have:

Special FAT Values

Value		Meaning
12bit	16bit	
0xFFF	0xFFFF	End of File
0xFF7	0xFFF7	Bad Sector
0x000	0x0000	Free Sector

The operating system uses the FAT to find where files are and to determine where it can place new files. When a new file is added to the disk, the operating system looks at the FAT to find the first FAT entry with a value of 0. If the amount of contiguous free space is insufficient to hold the file, then the operating system will look for the next available free space(thus creating a fragmented file). This process continues until enough clusters have been allocated to hold the file. Based on this information, the operating system creates a FAT chain for the file and makes the appropriate entries in the FAT.

Each file on the disk has an associated FAT chain. The FAT chain is the structure that the operating system uses to locate all of the clusters associated with a given file. For any given file, the value of each entry in the FAT "points" to the next cluster belonging to that file(the next cluster in the FAT chain).

So if a file occupied clusters 10, 11 ,12, 16, and 22. The files FAT chain would look like this: FAT[10] = 11; FAT[11] = 12; FAT[12] = 16; FAT[16] = 22; FAT[22] = 0xFFFF. The process of going from FAT entry to FAT entry until you reach the end of the file is called "walking the FAT chain".

We now know how to walk the FAT chain, to locate all of the clusters that belonging to a file. But where do we start our walk? The answer to that resides in the directory sectors.

The Directory Sectors

The last special control structure used by the disk operating system is the directory sector. The are two types of directory sectors: the root directory and subdirectories. The only differences between the root directory and subdirectories are in expandability and location. The actual structure of the directory entry is identical in the root directory and subdirectories.

The root directory is a control structure that lies outside of the "data" area and is not expandable. When you partition your disk, the maximum number of root directory entries is set and recorded in the boot sector. On the other hand, subdirectories are really a special kind of file with their own FAT chain. Space for subdirectories are allocated as needed, lie within the data storage area, consume available disk space, and are expandable. Although some versions of TOS limit the number of files in a subdirectory that can be viewed from the desktop window, there is essentially no limit of the number of files you can have in a subdirectory.

The critical structure within a directory sector is the directory entry. Each directory entry consists of 32 bytes and contains all of the information about the file. The elements of the directory entry are:

Directory Entry Contents

Element	Offset	# of Bytes
Filename	0x00	8
Extension	0x08	3
Attributes	0x0B	1
Reserved	0x0C	10
Time	0x16	2
Date	0x18	2
First cluster	0x1A	2
File Size	0x1C	4

The first field contains the filename. This name generally consists only of uppercase ASCII letters and numbers. If there are less than eight characters in the file name the remaining places are filled with the ASCII character 0x20 (space).

The filename extension occupies the next three bytes and is filled with spaces if necessary. Not that the dot (.) in a file name is not stored in the directory sector but is added by the operating system.

So the filename READ.ME would be stored as READ(space)(space)(space)(space)ME(space). If the first character of the filename is 0xE5 this means the file has been deleted.

The next field is the file attribute byte. It contains a bit code of the attributes of the file. The meaning of the bits are:

File Attributes

Bit	Value if set	Meaning if set
0	0x01	Read Only
1	0x02	Hidden File
2	0x04	System File
3	0x08	Volume Entry
4	0x10	Directory
5	0x20	Archive Bit
6		Reserved
7		Reserved

The value that is recorded in the directory entry is the result of "ORing" all of the applicable attribute bits together. So the attribute byte for a read-only hidden file that needs to be backed up(archive bit set) would be 0x01 | 0x02 | 0x20 = 0x23. A normal file has a file attribute of zero. The next ten bytes of the directory entry are reserved for future use.

The next two entries are the file creation/modification time and date. These entries are stored as bit encoded(to save space) Intel format words. The next entry contains the answer to the question we asked when discussing the FAT: the first cluster of the file. This is stored as an Intel encoded word and points to the first cluster in the file's FAT chain. The last element of the directory entry is the file size. The file size is 4 bytes long and is stored as an Intel format long word

In every subdirectory there are two special entries that are always the first two entries in the directory: "." and "..". The entry "." represents the current directory and contains all of the location information for the current directory. The entry ".." represents the parent directory(directory that contains the current subdirectory) and contains all of the location information for the parent directory.

Space for subdirectories is allocated one cluster at a time. As mentioned before, each directory entry take 32 bytes of disk space. So when **(#of bytes per cluster/32 bytes per directory entry)** number of files have been added to the subdirectory another cluster needs to be allocated to the subdirectory before any more files can be added.

The number of directory entries per cluster is 32 for a GEM partition with 512 bytes per sector. The FAT chain for the subdirectory is updated and the subdirectory can now hold another 32 entries. This process of expansion is essentially limitless(subject to disk capacity).

It is important to note that the very process of expansion creates a special kind of disk fragmentation: Directory Fragmentation. As the directory is expanded, the added directory clusters are never contiguous. This means that the operating system has to move the disk head all over the disk just to get information about a single subdirectory. Significant levels of directory fragmentation seriously impacts disk performance. You can detect directory fragmentation by observing the order that the disk fragmentation map is generated.

Causes of Disk Damage

Damage to data on your disks can result from hardware failure, magnetic media errors, software bugs, computer viruses, human error, or acts of nature.

Hardware

Hardware is those parts of the computer that you can see and touch such as the keyboard, mouse, disk drives, and the computer itself. Generally when a problem is attributed to hardware failure one or more of peripheral systems has failed. Most peripheral device failures do not result in data loss. They are generally restricted to the minor inconvenience of a trip to the repair technician.

However, if the peripheral that failed is a disk drive the potential for data loss is much greater. We need to distinguish between two types of drive failures: permanent and temporary. A permanent drive failure means that nothing can be done to recover the data because the drive cannot be accessed. The read/write heads not moving is an example of this. This is much different than a temporary failure such as the destruction of the disk partitioning information. With the tools provided by Diamond Edge, you have a very good chance at full data recovery.

Magnetic Media

Magnetic media is the magnetic disk material that is used to store your information. Media errors occur when there is an area of the disk that does not hold the magnetic charge. Weak or damaged disks often function completely normally until you attempt to read or write data to the damaged areas. This can lead to an error condition known as flipped bits. A bit can be a 1 or a 0. If the magnetic media is weak, then a bit can be a 1 the first time it is read and change to a zero the next time it is read. Use the Map Bad function with multiple passes to detect and mark bad sectors as unavailable for use.

If you detect bad media on a floppy disk, copy the remaining information to another disk and throw the bad disk away. Once a disk media weakens you will continue to experience data loss, often unrecoverable. You data is worth more than the cost of a new floppy disk.

Software and Viruses

Software related problems consist of "bugs" in user applications, conflicts between concurrently running applications, or computer viruses. Bugs are defects in the application software that can potentially lead to disk damage. If a program "crashes" with one or more bombs while writing to the disk, it can leave the disk in a corrupted state. If this occurs, it is advisable to run Disk Medic immediately to assess any possible damage.

Conflicts between concurrently running programs is generally between the applications and one or more Auto folder or accessory programs. Random crashes can occur in an application program that are actually the fault of bugs in an auto folder or desk accessory. If you suspect a conflict, reboot your computer with no auto folder or accessory programs active. Then add one auto folder program at a time, rebooting in between, until you identify the conflict. A binary search will find the offending auto folder program more quickly, but the one at a time method is more conservative. If a conflict causes a crash you should run Disk Medic to assess any possible damage.

Computer Viruses are another type of software threat to your disks. Viruses enter your system through boot sectors of floppy disks or attached to otherwise benign application programs. They become active and spread throughout all of your disks each time the program or triggering mechanism is run. Viruses can be benign or malignant.

Benign viruses often do little more than display a message on your screen at some predetermined time. However, malignant viruses can cause strange system behavior and even wipe out your entire disk structure and all of your files. It is strongly advisable to check every new floppy disk that enters your disk drive for viruses and keep all of your floppy disks write protected. Also, link viruses can be attached to actual programs and data files. Whenever, a new program is introduced to your disk it should be scanned for link viruses. This includes commercial software and software downloaded through a modem.

Oregon Research provides an excellent Virus protection program called the Ultimate Virus Killer for only $29.95. It has the capability of detecting and killing every known strain of virus on the Atari ST. It pro-

vides protection against unknown viruses and can kill many types of link viruses. These are particularly nasty viruses that attach themselves to programs and can cause significant disk damage. It repair damage to over 500 different benign executable boot sectors.

Computer viruses should be taken very seriously. They can easily cause 100% loss of data.

Human Error

Absolutely the most common cause of data loss is human error. These can result from overwriting a newer version of a file with an older version, accidently deleting an important file, or performing a disk operation that you didn't really want to do. Diamond Edge provides many tools to assist you in recovering from human induced loss of data.

Acts of Nature

If the power goes off or the electrical current is not consistent while performing disk read/write operations, you will likely have corrupted data or disk structure. Never operate you computer when there is an electrical storm in your general vicinity. If you can see it or hear it, shut down immediately.

Always ensure that you have a stable power source. Installing surge protectors and line noise filters is an excellent protection against this type of data loss. Note that surge protectors on your electric line is not always enough. If you have a modem connected to your computer then you also need a surge protector on your phone line. Many a "surge protected" computer system has been fried by surges coming over the phone lines from lightening strikes.

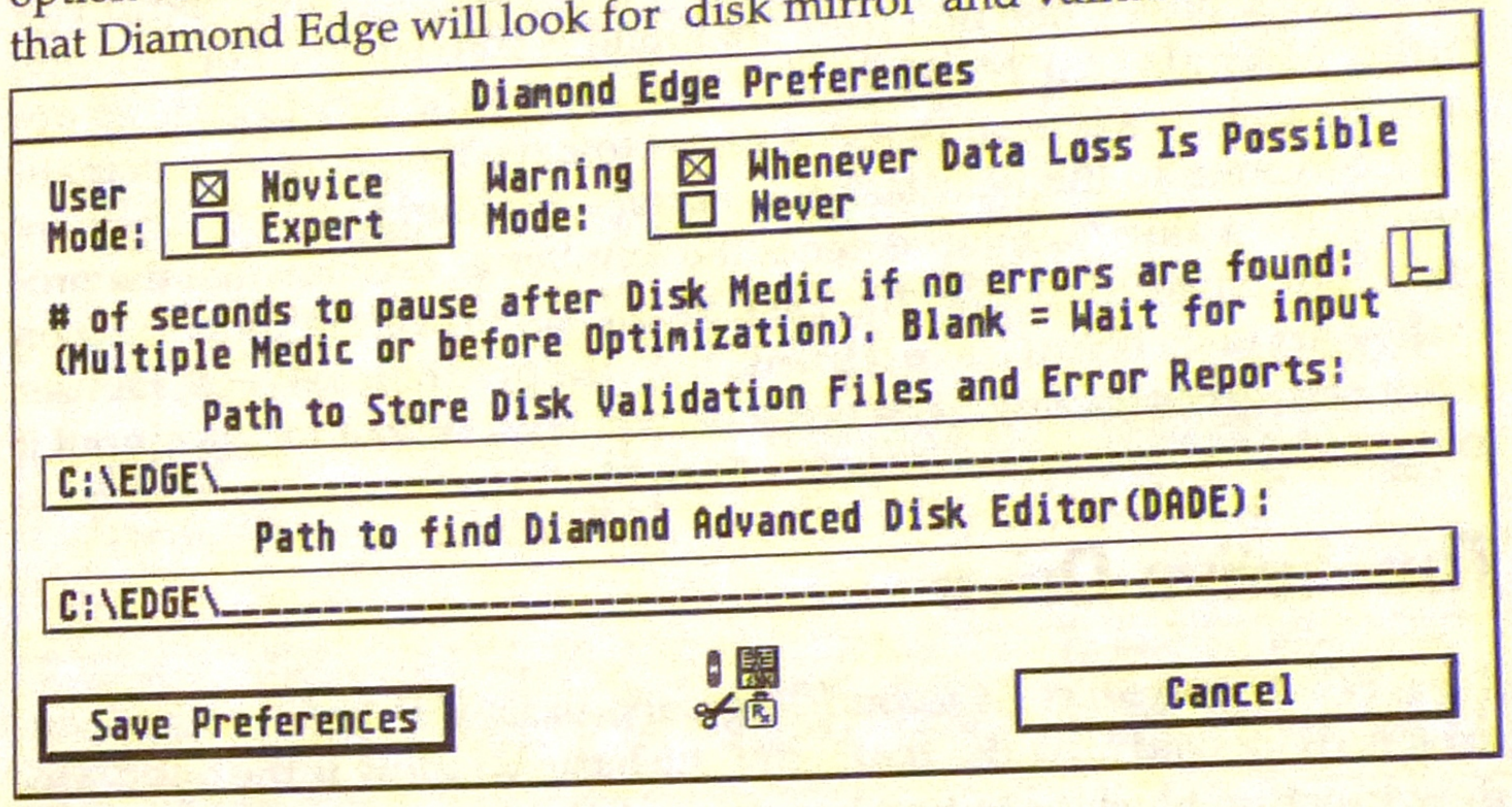
The disk drives in your Atari ST are as amazing and mysterious as any part of your computer. This primer to Atari ST disk structure was not intended to be an exhaustive treatment of the subject. We hope that it has given you a greater appreciation for the complexity of disk drive structure and provides enough of an understanding of your disks to get full utilization of the power of Diamond Edge.

If nothing else, we hope that this greater understanding of disk structure convinces you of the high probability of eventually experiencing disk structure corruption. The best protection against disk structure corruption are regularly updated disk information and mirror files and a current up to date hard disk backup.

3. Using Diamond Edge

Preferences

User selectable preferences are available through the Preferences option under the File menu. The selectable options include the paths that Diamond Edge will look for disk mirror and validation files.



✔ Note that all of the currently selected menu options and switches are saved in the configuration file. In addition, you have the option of selecting two very important options that control what types of operations a user may perform and what the level of warning that will be given before operations are executed.

User Mode

✔ The User Mode defines the level of experience with disk functions that YOU say you have. There are two levels: Novice and Expert. Although, all program functions are available to all users, additional warnings and explanations of the effects of the selected action will be given to users that select **Novice** user mode. If you select **Expert** user mode it is assumed that you know what you are doing when you select an action. Be certain that you do.

Warn Mode

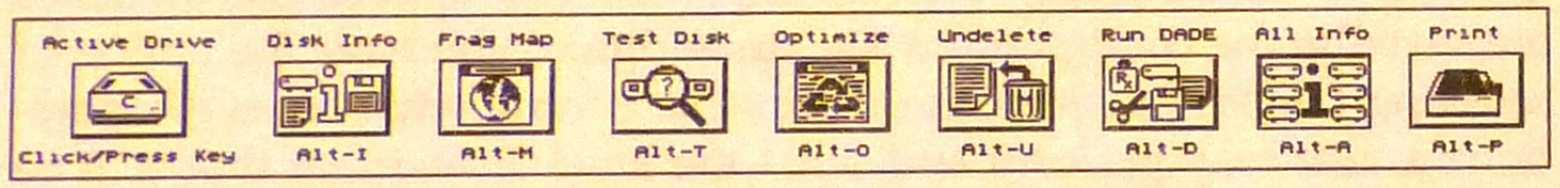
✔ Warn mode effects the amount of warning that you will receive before certain types of actions. There are two types of warning modes: Only before potential loss of data or Never. Note that even if you select Expert mode and Never warn you will still receive warnings before certain types of actions like repartitioning your hard disk or zeroing Drive C. Until you are very comfortable with the operation of Diamond Edge, it is strongly suggested that you select Novice User Mode and Always Warn options.

Disk Medic Pause with No Errors

There are several situations where automatic continuation after a Disk Medic pass when no errors are found is convenient. For instance, during Multiple Disk Medic pass or the Disk Medic pass prior to an optimization, you will find that it is easier to have the program automatically continue to the next task if no errors are found. If no errors are found, the value that you enter is the number of seconds that the program will pause to display the Disk Medic Report before continuing on to the next task. If you leave it blank, then the program will wait for user input before continuing. The program will always wait for user input if any disk errors are found during a Disk Medic pass.

The Active Drive

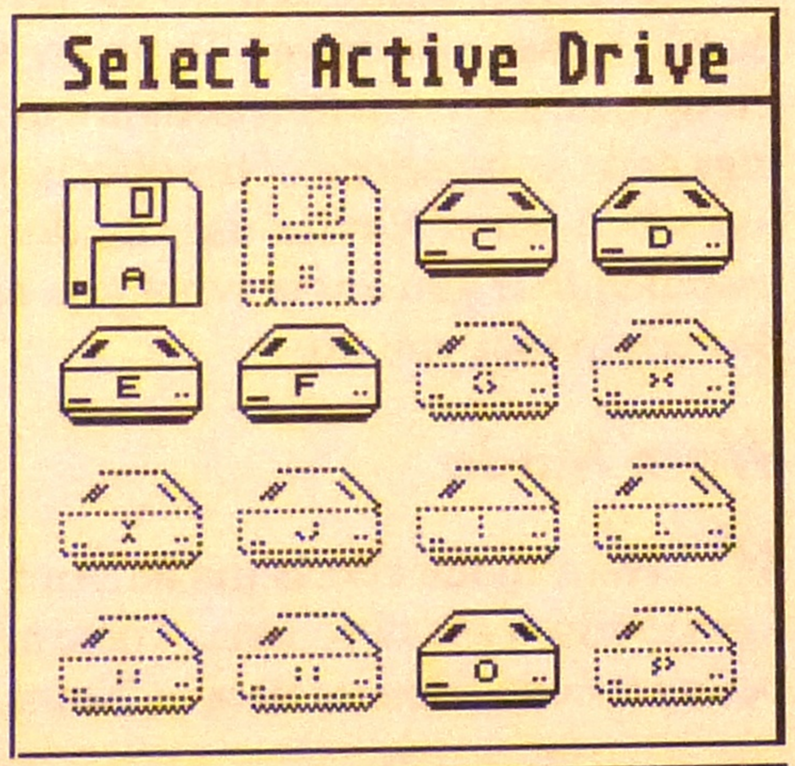
When you start Diamond Edge, the basic disk information window is displayed. At the bottom of the basic window is the basic window function block or button bar shown below.



There is one operating principal that applies to nearly every program function: the **Active Drive**. The basic window function block contains a set of buttons that operate on the **Active Drive**.

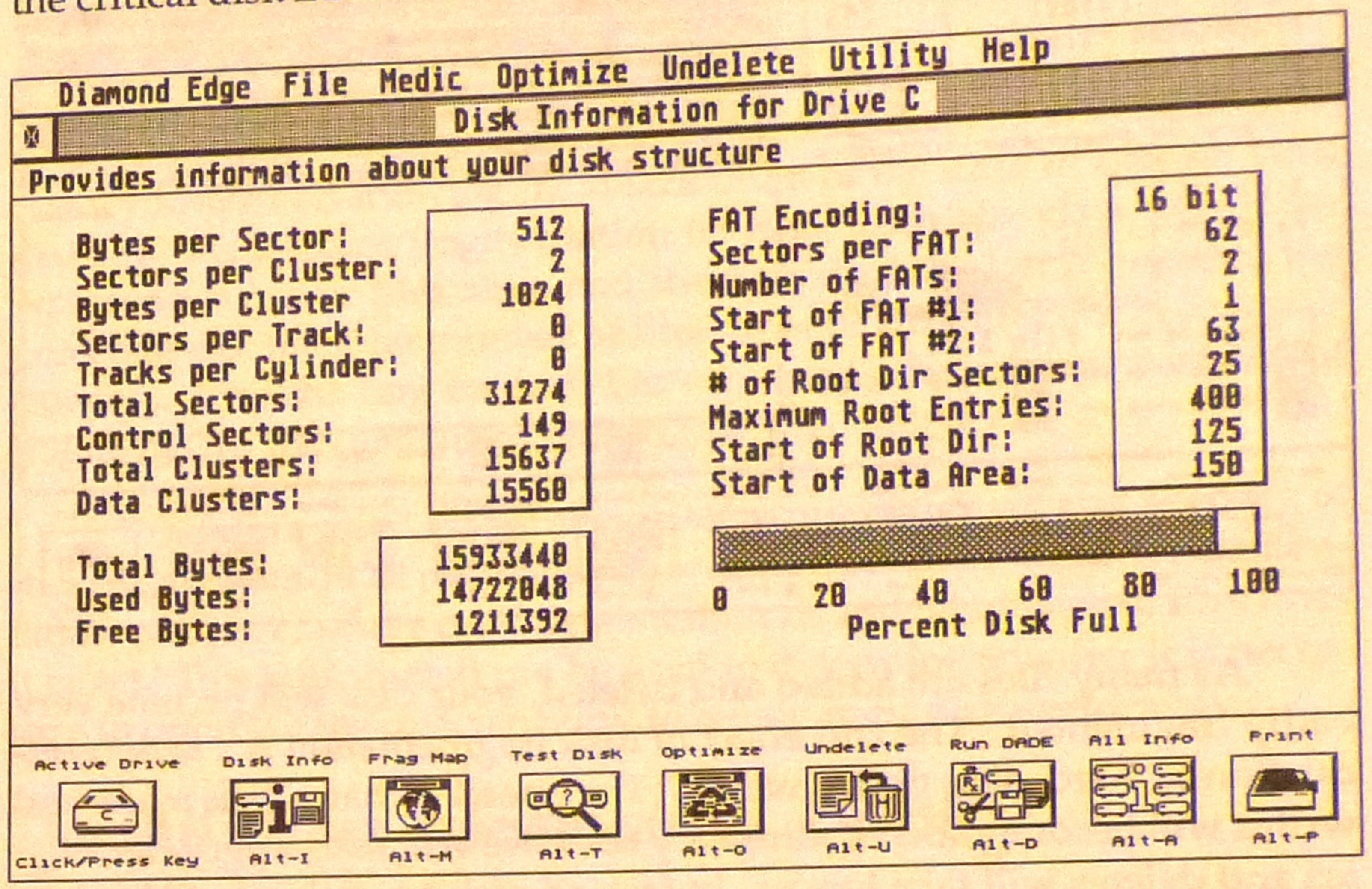
✔ **ALL PROGRAM FUNCTIONS ARE PERFORMED ON THE ACTIVE DRIVE.** The only exception to this rule are the SCSI level operations and Disk to Disk copy operations.

To change the Active Drive click on the drive button and select the drive you want to become the Active drive. You may also press the letter of the drive that you want to become the Active Drive.



The Disk Information Display

When you select **Disk Info** from the basic window function block, the critical disk information is displayed for the Active Drive.



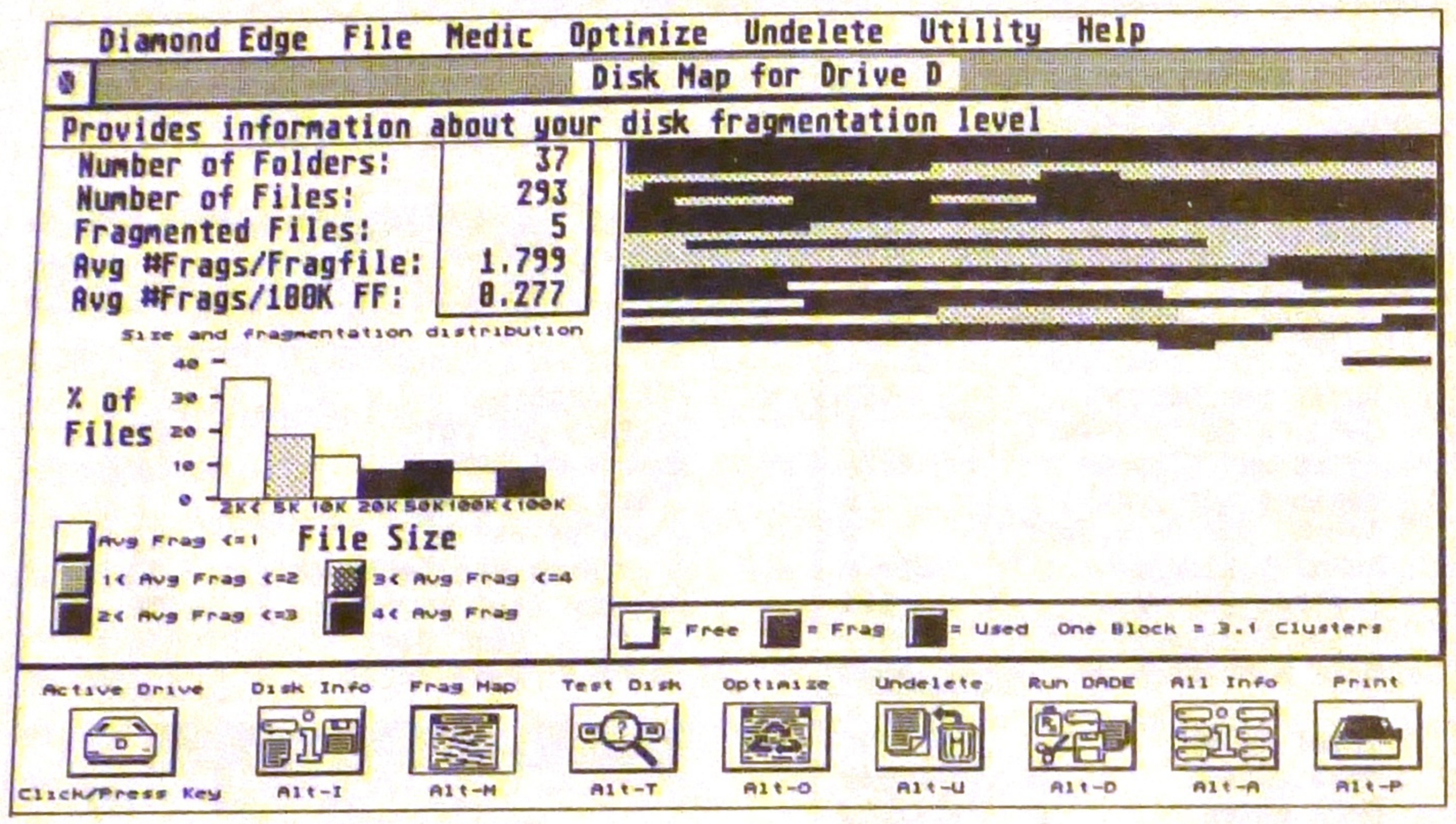
For a complete description of all of the items contained in the Disk Information Display, please refer to Chapter 2.

The Fragmentation Map Display

Selecting **Frag Map** from the basic window function block will create a visual display of the fragmentation level of the active drive and provide additional active drive statistics. Fragmentation occurs when parts of files become spread across your disk in different locations. This is a normal by product of the everyday process of creating files, deleting files, and then adding more files.

When a file is created/added, the operating system looks for free space on your disk to place the new file. When it finds some free space, it allocates the space to the new file. If the free space is not large enough to hold the new file, the operating system looks for additional free space in a different location. The result is a fragmented file: part of the file is placed in one location on the disk and part of the file is placed in a different location. Files will be split into as many fragments as are required.

Disk Fragmentation Display



As many files are added and deleted, your disk will become very badly fragmented. The end effect of disk fragmentation is a serious reduction in hard disk performance. The speed of hard disk reads and writes will become slower, directory searches will take longer, file copies and deletes will take longer. In fact, all disk operations will become significantly slower. The degradation in performance occurs slowly over time and you may not even notice it. Disk performance may not be much slower today than it was yesterday, but it's likely to be a lot slower than it was a month ago!

A very unique feature of Diamond Edge's fragmentation map is the way that the visual display is generated. The disk fragmentation map is generated in the order that an actual directory search would take. What you are witnessing is not only the level of file fragmentation on your disks, but also the level of directory fragmentation. If you have a heavily fragmented disk, you might notice that the map seems to fill in at seemingly random locations. This is demonstrating how hard your disk has to work to perform a simple directory search.

The Disk Fragmentation Map displays a number of useful summary statistics and graphs that allow you to assess the fragmentation level of your disk. For the purposes of the statistics 1 fragment is defined as an interruption in consecutive location of disk clusters in a file. So a file that is broken into two pieces on the disk has 1 fragment, a file that is broken into three separate pieces on the disk has 2 fragments, etc. A file that has all of it's clusters consecutive on the disk has no fragments.

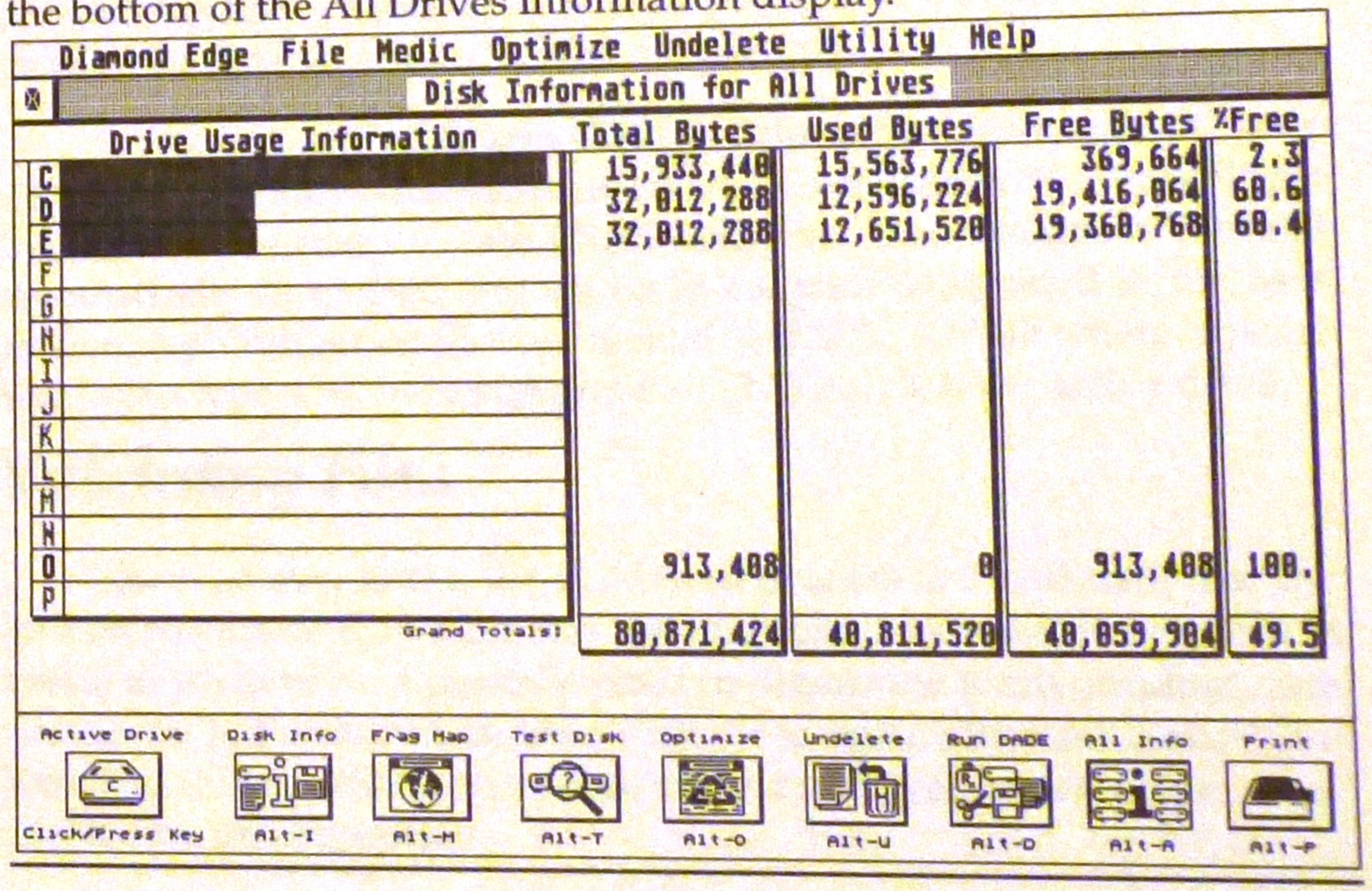
The summary statistics shown include the total number of folders on your disk, the total number of files on your disk, and the total number of fragmented files on your disk. Additional information is provided for the fragmented files showing the average number of fragments contained in each fragmented file and the average number of fragments contained in 100K bytes of fragmented files.

The graph in the Disk Fragmentation Map display is a relative frequency histogram showing the distribution of the sizes of files on your disk and the average fragmentation level for each size class. Each bar represents a bin of files sizes and the height of the bar represents the percentage of the total number of files on your disk that are in that size category. The shading on each bar represents the average number of fragments for files contained within that size category.

This information is useful in determining the level and severity of disk fragmentation. It provides you with an idea of how much fragmentation you have, where the fragmentation is, and what type of files does it effect. This information can be used to determine whether it is necessary to optimize the disk partition to regain hard disk performance.

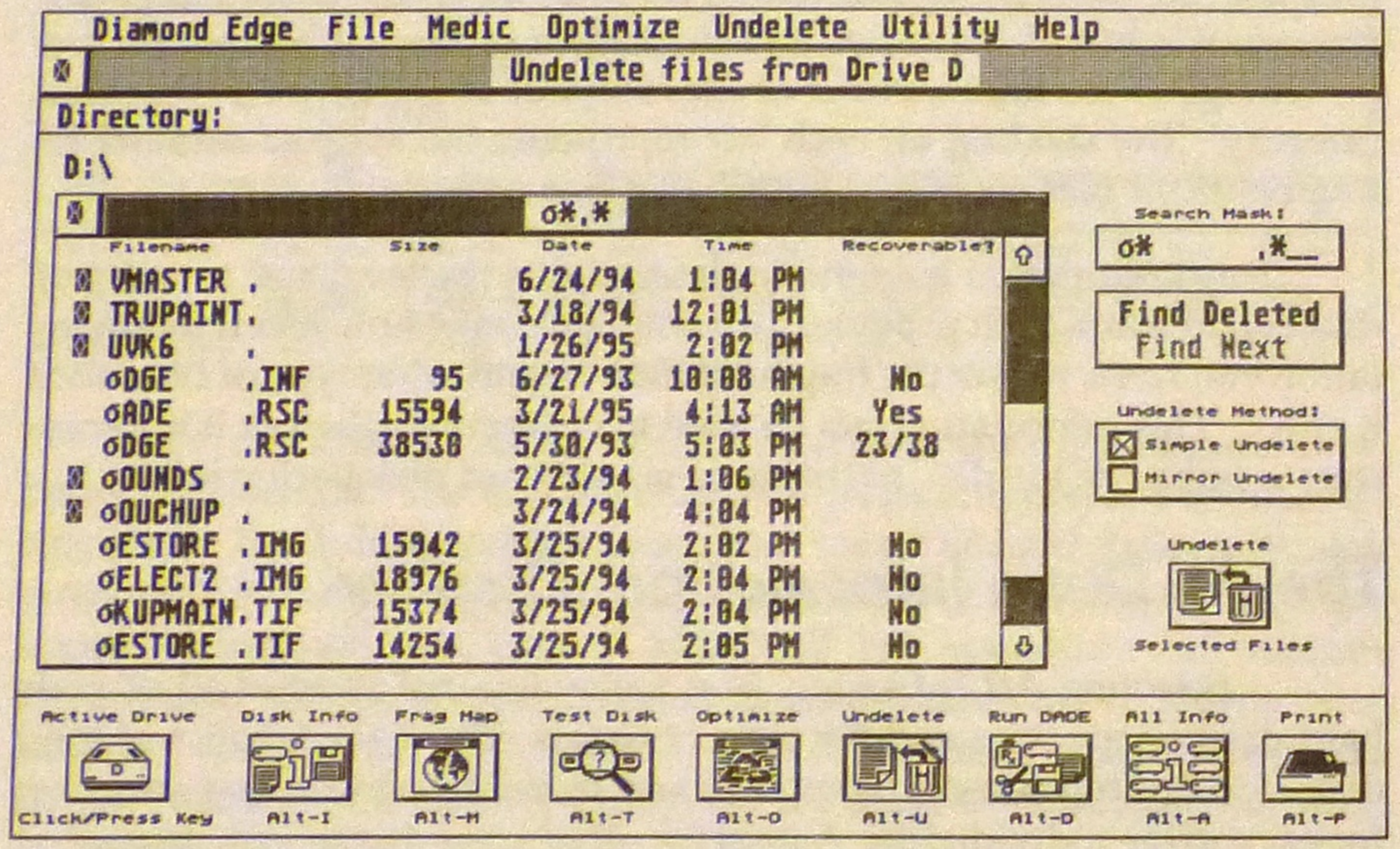
The All Drive Information Display

Selecting **All Info** gives you a visual description of all of your hard drives capacities and the current usage. The disk statistics of total disk capacity, bytes used, bytes free, and percentage bytes free are given for each drive individually. A total for all of your drives is displayed at the bottom of the All Drives Information display.



Undelete Display

One of the primary functions of Diamond Edge is the recovery of lost data. The easiest case is where the information regarding the file remains in the disk directory structure, but id hidden from normal viewing. The Undelete display is just like a normal file selector, except that the only files shown are those that have been deleted. Refer to Chapter 6 for a complete treatment of this important function.



4. Disk Diagnostics and Repair

One of Diamond Edge's primary functions is to protect against, identify, and repair damage to your disks and files. A wide variety of techniques are provided to assess state of, and to repair your disks. Each technique is designed to identify particular types of disk or file damage. The regular use of all of these techniques will provide you with excellent protection against data loss and early warning of potential trouble.

File Validation

Creating Validation Files

The first disk protection technique is file validation. The process of file validation involves the creation of a reference validation file used to compare the contents of files at a later date to determine if any data corruption has occurred. This is a file that contains calculated numbers that uniquely identify the contents of each file on the active disk and all of the file attributes(name, date, time, size, etc.). Diamond Edge provides both 32 bit Cyclical Redundancy Check(CRC) and 16 bit Checksum validation methods.

✔ When creating a validation files it is important to note that Checksums are significantly faster than CRCs and provide more than sufficient security for most applications. It is possible, although extremely unlikely, that a corrupted file would not be identified because it produced the same checksum as the uncorrupted reference. If absolute validation integrity is required, then the CRC method will provide an extra measure of security at the expense of some speed.

To create a CRC validation file from the active drive, select **Create CRC File** from the Medic Menu, to create a Checksum validation file for the active drive select **Create Checksum File**. The validation file will automatically be created and stored in the path designated in the user preferences. Validation files are named VALIDY_X.VLD where **Y** is the validation type (C=Checksum and R= CRC) and **X** is the active drive.

Validating Files

The next step in the file validation process is to validate that the files on the active disk have not become corrupted over time. This is useful to perform on a periodic basis to determine if any random data corruption has occurred on your disk. It is especially useful after you have had a major disk corruption/crash to determine what files were effected by the corruption.

To perform a validation of the active disk select **Validate Files** from the Medic Menu. A standard GEM file selector will appear and ask you for the reference validation file to use. When you have selected the validation file you will be asked what action to take when the program encounters a file that is not in the reference validation file. You have the option of either automatically adding the new file to the existing validation file, skipping all new files, or being prompted for action each time a file not contained in the reference file is encountered.

✔ At anytime during the creation of a validation file or the validation of files you may pause the process to study the screen by selecting **Control-S**. To restart the process, select **Control-Q**. You may abort the entire process by selecting **Control-C**.

What to do if a Validation Error is found

✔ A file validation error occurs when the calculated CRC/Checksum value does not match the value stored in the reference validation file. Each time this occurs you will be notified on the screen and the effected file will be recorded in a file named VALID_X.ERR where X is the active drive.

✔ If a validation error is recorded DON'T panic. After the validation is complete, view the effected files recorded in the validation error file. Even though a validation error is observed, you may not actually have a corrupted file. Many programs(especially auto folder programs) embed configuration information within the programs themselves. When they self modify to update the configuration information, these programs often do it in a way that doesn't change the files size, date, or time stamp. However, the bytes within the program has changed and this will trigger a false file validation error. Other types of files that modify themselves without updating the size, date or time stamp include some database files and files from the Atari Resource Construction program.

Testing Your Disk Structure

Diamond Edge provides the most comprehensive set of disk diagnostic and repair tools for your Atari ST. Diamond Edge has the capability to automatically detect and correct any type of disk structure problem and return even heavily damaged disks to full functionality.

✔ As with all of Diamond Edge's functions, the Disk Medic can be performed on either floppy disks or hard disk partitions. Once a disk structure becomes corrupted, it can quickly develop into catastrophic data loss. We strongly recommend that you perform run Disk Medic on ALL or your hard disk partitions at least once a week.

Diamond Disk Medic will test and correct(if Fix is selected) the following types of disk structure errors:

- **Validates the Boot Sector:** This sector tells the operating system what the disk structure looks like and where to find things. If the boot sector is invalid, then you will have to reinstall a valid boot sector using the Archive function before you can continue.
- **Validates that FAT #1 equals FAT #2:** If FAT #1 does not equal FAT #2 then you will be given the option to use one or the other FAT for the Medic pass. FAT means File Allocation Table, please see section 8.3.2 for a full description.
- **Invalid Directories:** These are directory sectors which are either improperly terminated(the next cluster entry in the FAT points to an invalid cluster) or the location where the directory is supposed to be is not a directory. Invalid directories are either truncated, if improperly terminated, or deleted if not valid.
- **Unreadable Directory Sectors:** These are directory sectors that could not be read due to physical disk damage. Readable portions of the effected directory sector chain are adjusted to eliminate the unreadable portion or the directory is deleted from the parent directory if no sectors are readable.
- **Illegal File Names:** These are files that contain unusual characters. Illegal characters are replaced with a X or left unchanged depending on user preference. An illegal file name error is also triggered if there are two or more files in the same directory with the same name.
- **Bad Directory Entries:** These are entries within a directory sector that cannot be recognized as being a directory entry. Bad directory entries are deleted.
- **Inconsistent File Size:** This is where the file size contained in the directory entry does not match the number of clusters allocated for the file on the disk. When fixed, a file with an inconsistent file size is adjusted to reflect the size it actually occupies on the disk.
- **Bad FAT entries:** These are entries in the FAT that point to illegal or nonexistent clusters. Files with Bad directory entries are truncated and their file size adjusted.
- **FAT chain collisions:** These are where two or more files FAT chains contain the same clusters. Independent copies of all effected files are made with valid FAT chains. The user will need to determine which of the effected files are corrupted by the collision.
- **Lost Clusters:** These are clusters that are allocated in the FAT, but do not belong to any of the files currently on the disk.

Disk Medic Options

There are several basic options, selectable through the Medic Menu, that control the action of Disk Medic. The first is **Auto-Fix** errors. If you select Auto-Fix errors, then Diamond Edge will automatically correct all disk errors found during the Disk Medic evaluation of the active drive. It will log all errors found in a error file MEDIC_X.ERR where **X** is the active drive. Disk Medic Error files are saved to the directory specified for Validation files in the Preferences menu. After the Disk Medic pass, you may view the error file for information on what errors were found and fixed.

✔ If you decide not to Automatically fix errors, then the Disk Medic will evaluate the active disk and record all errors found in the active drive error file, but will not fix any of the errors. You may then view the error file from the Disk Medic Report to find out what errors were found and what files are effected. At this point you may either cancel without fixing the errors or instruct the Disk Medic to fix the errors.

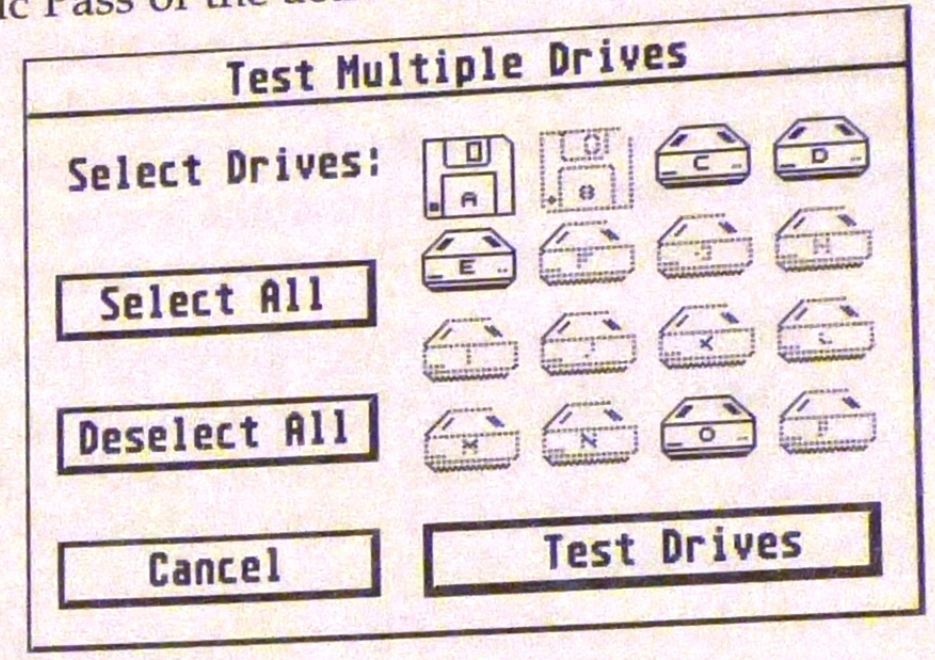
One of the disk error types that the Disk Medic can fix is illegal file names. Filenames that contain characters **other** than A-Z, 0-9, or _ can cause problems for some programs. However, some people cherish the ability to create file names like ©1992.ß1 or !READ.ME!. You can have the Disk Medic ignore illegal file names or have illegal filename characters replaced with **X**, so **!READ.ME!** would be changed to **XREAD.MEX**.

When Diamond Disk Medic performs it's analysis of your disk, it associates all of the File Allocation Table(FAT) entries with files contained in the directory structure. Clusters in the FAT table that cannot be associated with any file in the directory structure are called Lost Clusters.

When you fix the disk errors you may select to either discard the data in the lost clusters or to save the data in the lost clusters to a file. When you select **Save Lost Clusters** from the Medic Menu, any lost clusters will be saved to a file LOSTCLUS.DAT when the active drive is fixed. If LOSTCLUS.DAT already exists on the active drive, then LOSTCLU0.DAT, LOSTCLUS1.DAT etc. will be created in succession.

✔ We highly recommend that you save lost clusters. You can examine the information in the lost clusters data file and often fully recover the lost information. The ability to recover text information form a lost cluster data file is especially good. Although the information you want may be present, non-text data is very difficult to identify and recover from a lost cluster data file.

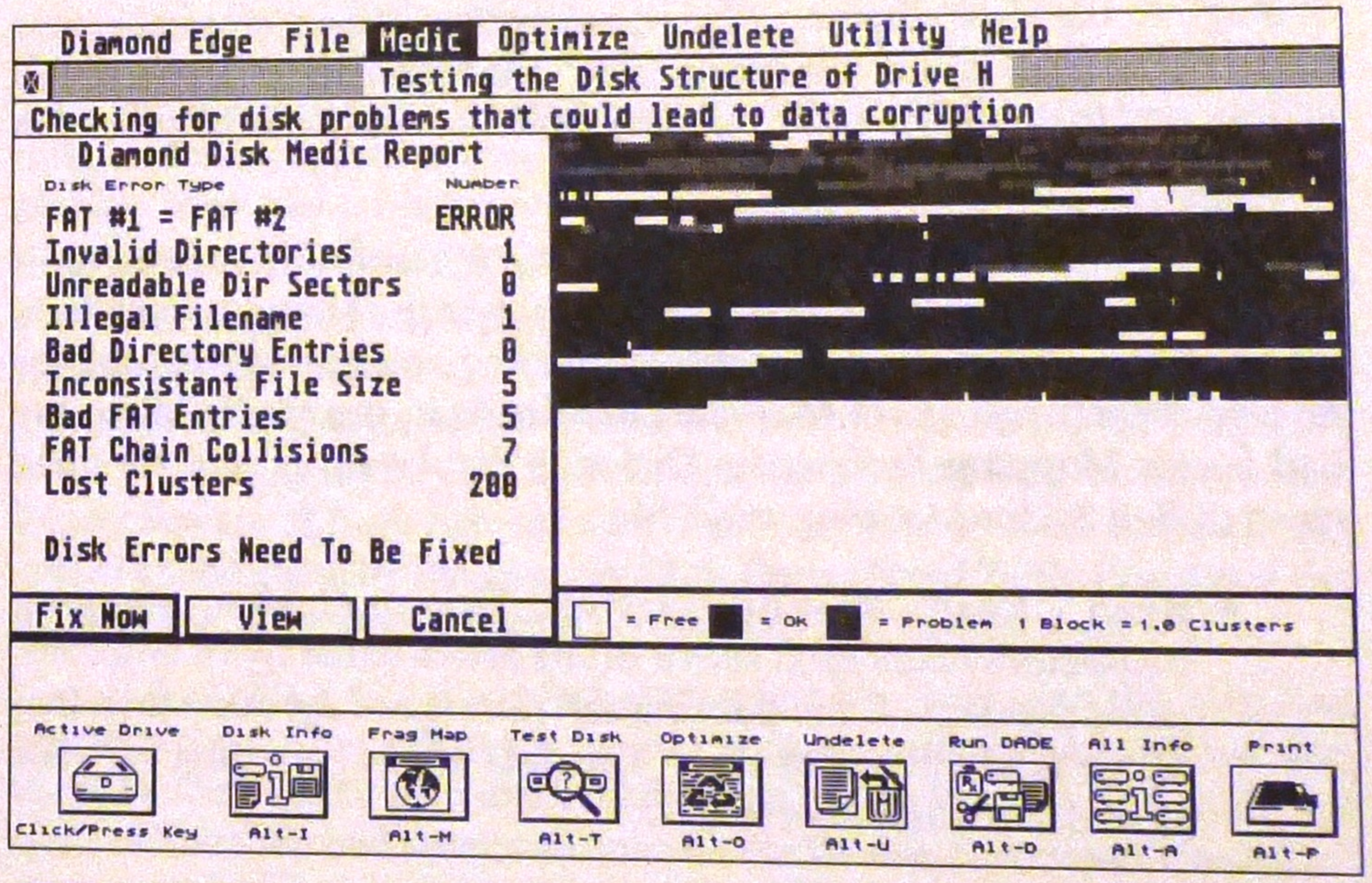
To begin the Disk Medic Pass of the active drive, select **Test Disk** from the Medic Menu. Remember that the drive you are testing is the Active drive. If you want to test more than one drive in a single pass select **Test Multiple** from the Medic Menu. The Test Multiple selection dialog will then be displayed. Select all of the drives that you want to run Disk Medic on and then select **Test Drives**.



Disk Medic Report

After you have begun Disk Medic on the active drive, the Disk Medic operation screen will be displayed. The program first validates the integrity of the boot sector and verifies that FAT #1 equals FAT #2. If FAT #1 does not equal FAT #2 then you will be given the option to use one or the other FAT for the Medic pass. If you have Auto-Fix selected, then you will be given the option to turn Auto-Fix off.

✔ When FAT #1 does not equal FAT #2, it is ALWAYS a good idea to run Disk Medic with Auto-Fix turned off using both FATs. Then you should examine the disk errors resulting from the use of each FAT and select the FAT that produces the least number of disk errors.



The Disk Medic pass then proceeds to examine the entire disk structure looking for errors. As the Disk Medic pass proceeds the location of the disk being examined is displayed in the graphic window. Areas where errors are detected are displayed in Red(cross hatch in monochrome). If you turn Visual Update off in the Medic Menu, then the testing process will be faster but you will not be shown the locations of the errors.

The Disk Medic Report gives the total number of each type of disk error, described above, that was found on the active drive. If Auto-Fix was selected, any errors detected were corrected during the Disk Medic pass. This is indicated by the **Disk Errors Have Been Fixed** message in the Report Screen. If Auto-Fix was not selected and errors were detected this would be indicated by the message **Disk Errors Need To Be Fixed** and the appearance of a **Fix Now** button on the Disk Medic report screen.

✔ Any errors detected and the files they effect can be viewed from the Medic Report screen by selecting View. If Auto-Fix was not selected and you want to fix the disk errors, then selected **Fix Now** from the Medic Report screen. This will launch another Disk Medic pass that will fix all of the disk errors. If you do not want to fix the errors now, then select **Cancel**. If no errors were found or after Disk errors have been fixed, select **Continue**.

Mapping Bad Sectors on Your Disk

The reliability of today's SCSI drives is improving every year, however they occasionally develop portions of the disk that become unusable. This may be due to weakened magnetic media or from physical damage to the disk surface. Regardless of the cause, bad sectors can result in loss of data and disk corruption. The Map Bad Sectors function is designed to identify areas over your disk that are potential problems and mark them as unusable.

You may either Map Bad sectors on just the active drive or on multiple drives. The status of the Auto-Fix option in the Medic menu effects whether bad sectors are automatically marked bad or just recorded for review. When you select **Map Bad Sectors** from the Medic Menu the Bad Sector Mapping Preferences Dialog is displayed. There are three types of Bad Sector Mapping available:

- **Read - Read Non-destructive:** This method of mapping badsectors reads each sector of the active drive twice and compares the data. If the information read is not the same then that sector is identified as bad. The data already existing on your hard drive is not effected.

- **Read - Write - Read Non-destructive:** This method of mapping bad sectors reads each sector of the active drive and writes it back to the hard drive. The sector is then read in again and compared to the data originally read. If the information is not the same then that sector is identified as bad. The existing data on the active drive is not effected.
- **Write - Read Destructive**: This method of bad sector mapping writes a random bit pattern to each sector on the active drive and then reads that sector back. If the information read back is not the same as the random bit pattern that was written then that sector is marked as bad. This is a destructive mapping and all data existing on the drive will be destroyed.

✔ These methods are listed in increasing order of effectiveness and increasing order of risk. Read - Read is the least effective method but no data is ever written, so even data in marginally bad sectors will not be lost. Read-Write-Read is the most effective non-destructive method, but it has a very small chance of one sector data loss if you have a bad sector where the data can still be read, but goes bad when written to. Write - Read is the most effective method of detecting bad sectors, but all data existing on the drive will be lost (hint: backup your data first with Diamond Back 3).

✔ REMEMBER TO TURN OFF ALL DISK CACHING BEFORE DOING A BAD SECTOR MAP OF YOUR DISK!

The effectiveness of all methods can be increased by performing multiple bad sector mapping passes. Many bad sectors are simply weak sectors that sometimes operate correctly and sometimes produce checksum errors when reading or writing. These types of sectors may not be properly identified as bad if you just perform one bad map pass. The number of passes that you want to perform is entered in the Mapping Bad Sectors preferences dialog. The time it takes to complete the process increases with the number of passes. When you are ready to begin mapping bad sectors, select **Map Bad Sectors Now**.

When the Map Bad pass is complete the logical sector numbers of each bad sector found is displayed in the Map Bad Report screen. If Auto-Fix was selected, any errors detected were corrected during the Map Bad pass. This is indicated by the **Disk Errors Have Been Fixed** message in the Report Screen. If Auto-Fix was not selected and errors were detected this would be indicated by the message **Disk Errors Need To Be Fixed** and the appearance of a **Fix Now** button on the Map Bad Report screen.

If Auto-Fix was not selected, then select **Fix Now** from the Map Bad Report screen. This will mark the bad sectors as unavailable for use and launch a Disk Medic pass that will fix all of the disk errors. If you do not want to mark the bad sectors as unavailable for use, then select **Cancel**.

Partitioning Your Hard Drive

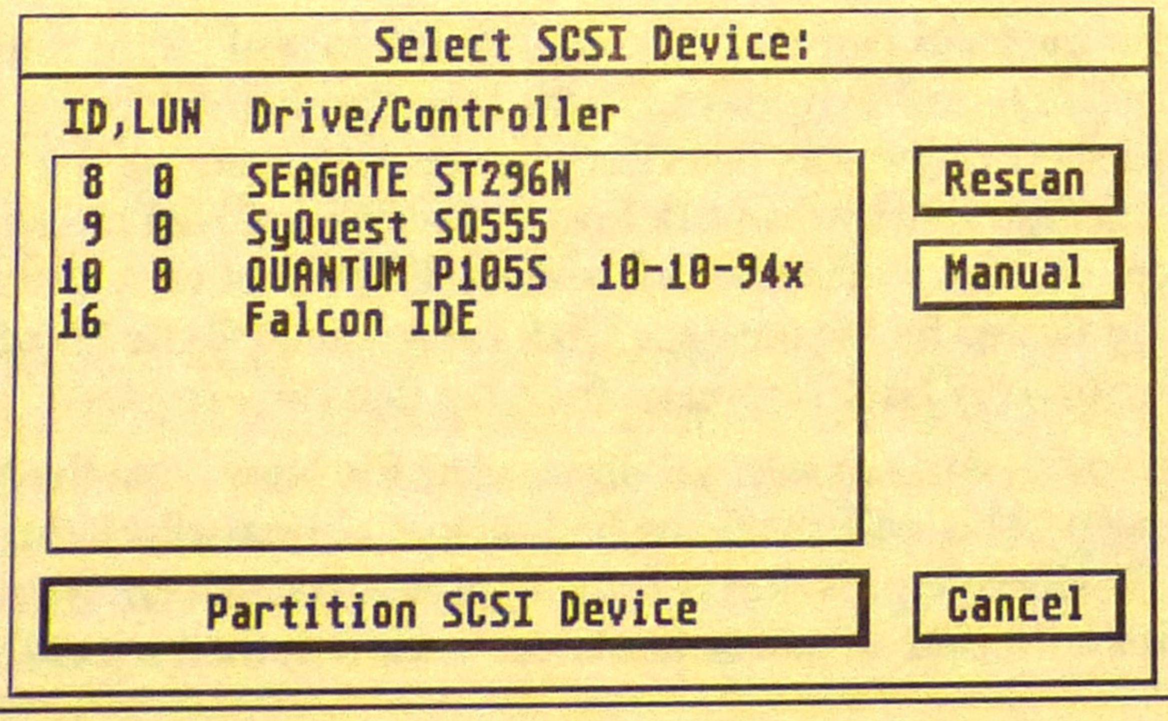
A critical component of the operation and function of your hard drive is the drive partitioning. This is how you have your physical disks divided into smaller logical disk drives that are commonly referred to as partitions. A single physical disk drive can have many logical partitions. In general, the performance of your disk can be improved by splitting your disk into a greater number of smaller partitions. There is of course a limit of 14 active partitions imposed by the operating system on all of your physical disk drives.

Diamond Edge provides a full featured disk partitioning system that supports both the Atari Disk partitioning standard as well as third party partitioning standards such as Supra, BMS, and ICD. You can completely change your disk partitioning scheme, just change one partition, hide and unhide secret partitions, save and restore partitioning information. The flexibility of creating a new disk structure or modifying old ones is unlimited.

Selecting the SCSI Drive to Partition

✔ It is strongly suggested that the reader consult Section 8 of this manual before attempting to use this function. This will give the reader the basic understanding of disk and partition structure necessary to fully utilize this function.

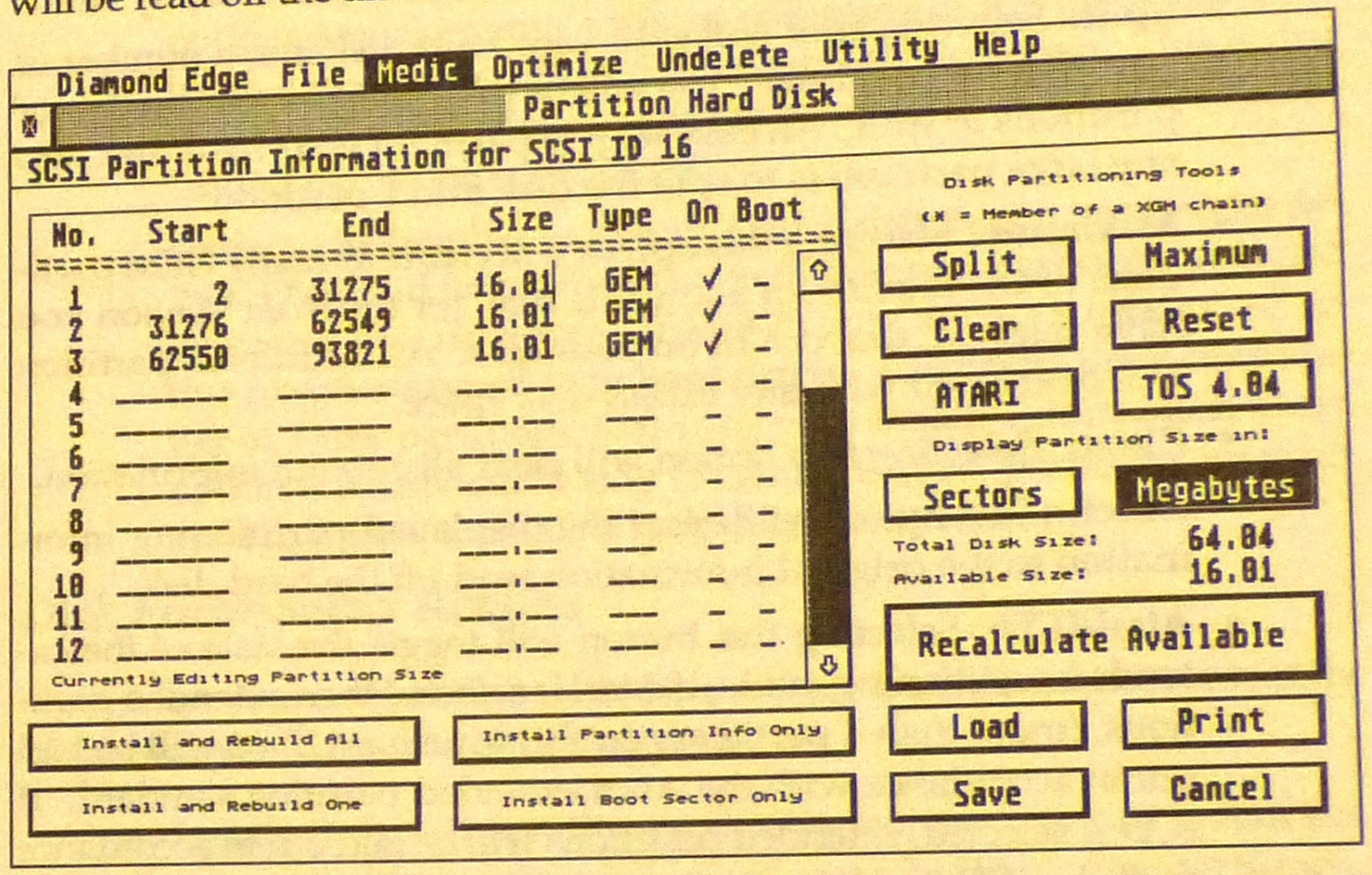
When you select **Partition Hard Drive** from the Medic Menu, Diamond Edge will scan ST ACSI id numbers 0-7 and TT SCSI and Falcon SCSI-2 id numbers 8-15 and Falcon IDE device 16 for active devices. The drive identification of the devices found will be displayed in the Select SCSI Device dialog. Select the SCSI device that you want to partition and then select **Partition SCSI Device**.



Some early SCSI drives and MFM drives attached to the ST using a SCSI-> MFM adapter(like the Adaptec 4000) will not respond correctly to the SCSI Inquiry command. If your drive does not respond to the inquiry select **Rescan**, it may just be slow to respond. If it still does not respond to the inquiry, then select **Manual** from the Select SCSI Device dialog. Select the SCSI ID number and Logical Unit Number(LUN) of the device that you want to partition. Select **OK** to partition the selected hard drive.

Disk Partitioning Options

After selecting Partition SCSI Disk, the current partition information will be read off the disk and the SCSI Partition Screen will be displayed.



If valid partitioning information exists on the disk it will be displayed as the basis to begin the partitioning session. There information displayed for each partition includes the starting and ending physical sectors, the size of the partition in physical sectors/Megs , the type of Partition GEM/BGM, etc., whether the partition is active, and whether the partition is bootable. The total size of the hard disk and remaining available space is also displayed.

There are three basic editing modes: Start/End sectors, Size, and Partition Type. The current editing mode is displayed at the bottom of the partition window. To change fields within an edit mode, you may use the usual Arrow and Tab keys or point the mouse and click the left button. To change edit modes click on the object that you want to edit with the left mouse button.

✔ As you edit your partitions there are two basic rules regarding the sizes and locations of partitions: 1) Partitions must be sequential i.e. the start of partition #2 is after the end of partition #1 and 2) You cannot create partitions whose total size exceeds the size of your hard disk. The following partitioning aids are provided to expedite the task of partitioning your disk:

- **Display Size Mode:** You can select to have the disk and partition size information displayed in either Sectors or Megs. Diamond Edge displays Megs consistent with the ICD definition Meg = 1,000,000 bytes. Note that this is different than the traditional definition of a Megabyte = 1,048,576 bytes used by Atari.
- **Split:** Selecting split will split your hard disk into a number of partitions of equal size. You will be prompted for the number of partitions to split. An easy way to partition your hard disk into one large partition is to split the disk into 1 partition.
- **Maximum:** Selecting this option will create as many GEM partitions of the maximum allowable size, for the Rom Version you have selected, that can fit on your disk. An additional partition will be created with any excess disk space.
- **Clear:** Selecting this option will clear all entered information.
- **Reset:** This option will reset the displayed partitioning information to the original information read off the hard disk.
- **Atari/ICD:** Selecting this button will toggle the state of the extended partitioning mode. If Atari is selected then extended partitions, (more than 4 partitions on the same hard disk) will be laid out in accordance with the Atari extended partition standard. If ICD is selected, extended partitions will be laid out in accordance with the ICD/Supra extended partition standard. Please see section 8.2 for a complete description of extended partitions.
- **TOSxxx:** Selecting this button will toggle the state of the Rom dependence mode. If TOS1.02 is selected, then partitions whether GEM or BGM will be limited to 32,767 clusters. These disks will be able to be read on any ST regardless of ROM version. If you have TOS 1.4 or above you can specify that partitions may include up to 65,535 clusters. These disks cannot be accessed by machines with TOS1.02 or 1.00.
- **Save:** You can save the hard disk partitioning information currently displayed to a file for future reference or installation. This option creates a SCSI disk information file equivalent to those created by the Save SCSI Information function of the Archive menu.

- **Load:** You can load previously saved partition information files by selecting this function. This is useful for partitioning a new disk "just like" another disk. The loaded information is fully editable.
- **Print:** Selecting this option will print a copy of the currently displayed partition information to your printer. The use of this option is **HIGHLY** recommended. It is **ALWAYS** a good idea to have a printed copy of the physical structure of your disk.
- **Recalculate:** At any time during your editing session you can recalculate the amount of available space on your disk. You can either click on the **Recalculate** button or press the **Return key** to update the available information. All of the currently selected options are taken into account and some of the partitions may begin in a slightly different place after recalculation. This is due to changes in the control sector requirements of the different extended partitioning protocols.
- **Active:** Remember to specify which drives you want to be active. Diamond Edge allows up to 64 different partitions installed on each disk. However, GEMDOS only allows a total of 14 active partitions on all your hard drives. Changing the Active status of extra partitions and using the Install Information Only option is a convenient way to create "Secret" partitions on your hard drive not accessible unless you reactivate it.

Disk Partitioning Actions

After you are satisfied with the partitioning information currently displayed you have four action options:

- **Install Only:** Select this option if you only want to install the current partition information without rebuilding any partitions. This is used to change the activation status of hidden partitions or to reinstall the partition information after a hard disk crash without changing any of the data on the hard drive.

✔ Having the partitioning information in physical sector 0 wiped out is a more frequent occurrence than most people might imagine. You should always have a printout of your partition parameters and a SCSI information file saved to floppy disk.

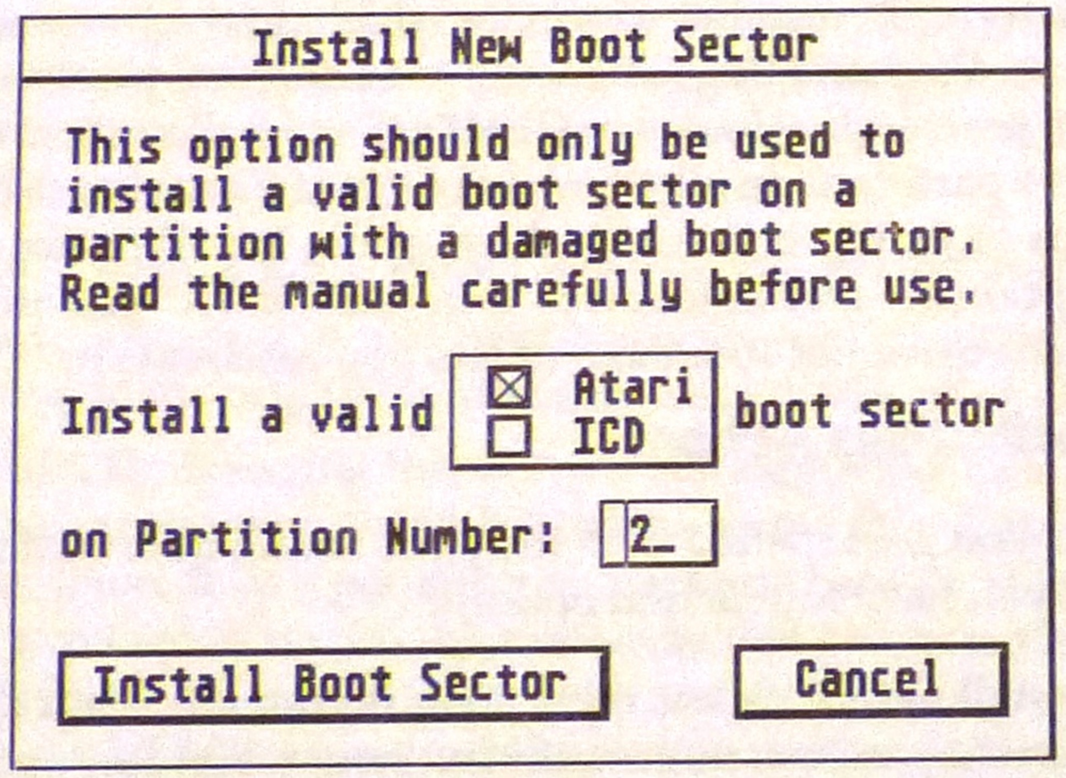
- **Install and Rebuild One:** This option will install the displayed partition information and rebuild one specified partition. Rebuilding a partition involves the creation of a boot sector, FAT sectors, and Root Directory sectors of the appropriate size on the specified partition and then zeroing them. All data in the area effected will be lost.

Disk Partitioning Actions(cont)

- **Install and Rebuild All:** Use this option to repartition your entire hard disk. The current disk partitioning information will be installed and all partitions will be rebuilt. All existing data on the entire hard disk will be lost.

✔ If you Accidently Rebuild a partition and need information that was previously in that area you can recover the information by reinstalling the old partitioning information and reinstalling the disk structure from an archived Disk Information File. See Chapter 7 Archiving Disk Information for instructions on how to perform these life saving actions.

- Install Boot Sector Only: This option should be used only to install a valid boot sector on a partition whose boot sector has been damaged.



Select ICD or Atari depending on which utility you used to format the drive. This effects the size of the logical sector sizes used to create a particular disk. ICD method allows larger partitions with smaller logical sector sizes than Atari. It is very important that you install the same kind of boot sector as was originally placed on the drive or you will not be able to access the information on the drive after this operation is complete.

- **Cancel:** If you have not performed any other action, then Selecting Cancel will return you to the Diamond Edge Basic Window. If you have installed partition information or rebuilt one or more partitions, then selecting **Cancel** will perform a cold reboot of your system. This is required for the operating system to become aware of the new disk structure.

5. Optimizing Your Drives

As time goes on, directories and files become spread around your hard disk in different locations. When you ask your computer to read or write data, it has to look harder and move farther to complete the request. This significantly degrades the performance of your hard disk. Disk optimization, also referred to as disk defragmentation, reorganization, and file remapping, involves the reorganization of files on a disk to improve the disk performance.

✔ File fragmentation occurs when parts of files become spread across your disk in different locations. This is a normal by product of the everyday process of creating files, deleting files, and then adding more files. When a file is created or added, the operating system looks for free space on your disk to place the new file. As soon as it finds some free space, it allocates the space to the new file. If the free space is not large enough to hold the new file, the operating system has to go looking for more free space in a different location.

The result is a **fragmented file:** part of the file is placed in one location on the disk and part of the file is placed in a different location. Files will be split into as many fragments as are required. Accessing a fragmented file is much slower than accessing an unfragmented file.

Subdirectories can also become fragmented. Directories are really just special kinds of files. When you create a subdirectory, one cluster is allocated to the directory. Since each directory entry consumes 32 bytes of disk space, this is enough space to store **(#of bytes per cluster)/32** directory entries.

For a standard GEM partition with 1,024 bytes per cluster, this number is 32. When you add more than files than the current directory can hold, the operating system allocates another cluster to hold additional directory entries. The additional cluster is never right next to the first cluster so the operating system has to go to many places on the disk just to display a directory.

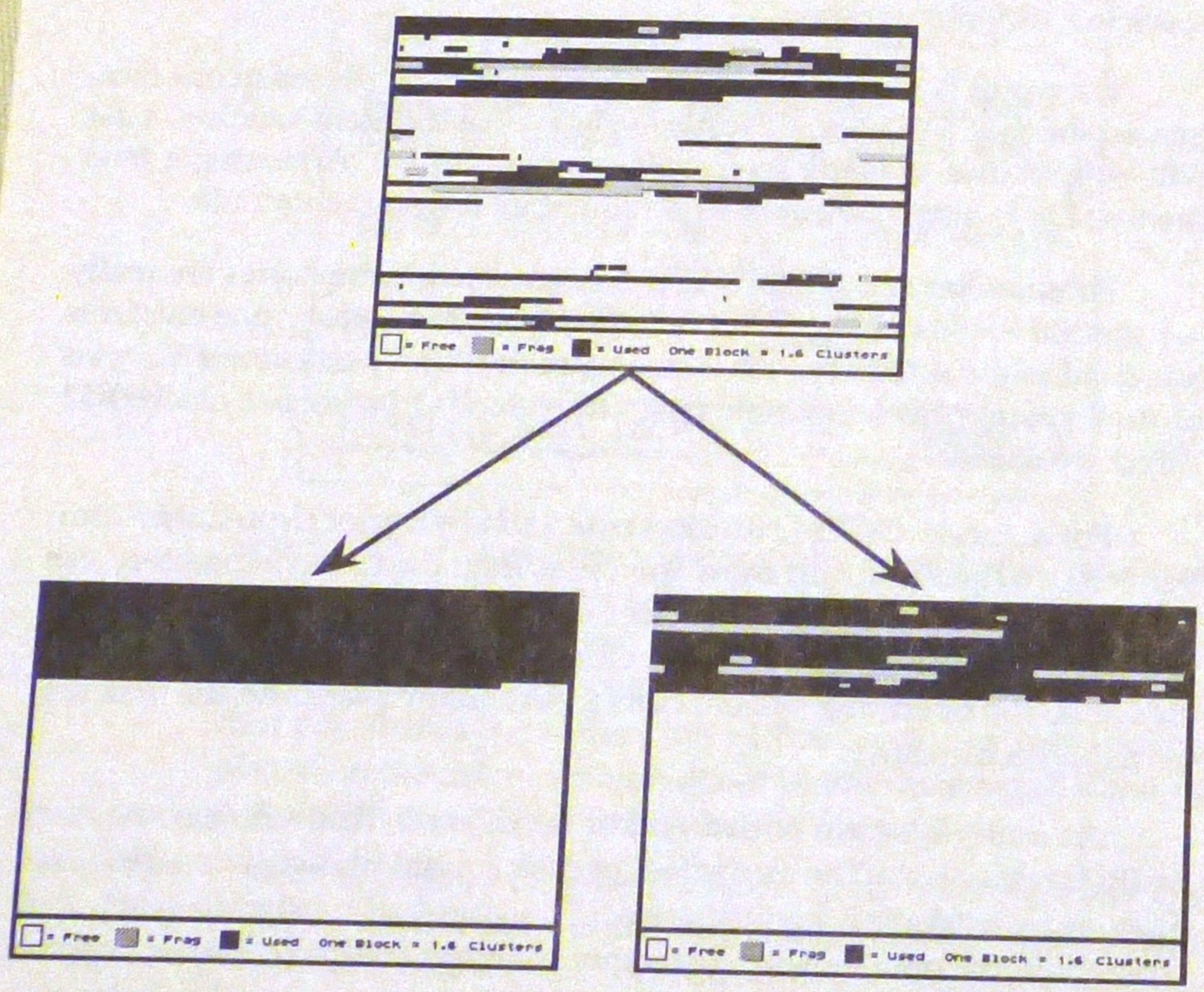
As many files are added and deleted, your disk will become very badly fragmented. The end effect of disk fragmentation is a serious reduction in hard disk performance. The speed of hard disk reads and writes will become slower, directory searches will take longer, copies and deletes will take longer. In fact all disk operations will become significantly slower because the disk head must travel to many locations on the disk to complete each read or write request.

✔ Disk head movement is the slowest component of any disk operation and the amount of disk head movement is critical to disk performance. The degradation in performance occurs slowly over time and you may not even notice it. Performance may not be much slower today than it was yesterday, but it's likely to be a lot slower than it was a month ago! Regular optimization will insure that your drives are always operating at their fastest possible speed.

Optimization Methods

Diamond Edge offers two optimization methods: **Full optimization** and **Compress Free Space**. You select the optimization methods by selecting the appropriate method under the Optimize menu. Depending on your individual requirements, both of these methods can be used effectively to help maintain your hard disk at peak operating performance. A graphic description of the two methods is shown below.

Source Partition



Full Optimization   Compress Free Space

Full optimization is a complete optimization of your hard disk. First, all of the directory sectors of a given subdirectory are consolidated, then all of the sectors in each file within the subdirectory are defragmented and consolidated, then all of the files within a subdirectory are located consecutively on the disk.

Actually it happens all at once, but the result complete and total optimization: Subdirectory clusters are contiguous, all of the file clusters are contiguous, and all of the files within a subdirectory are located consecutively on the disk. Diamond Edge's state of the art disk optimization algorithm is the fastest, by a significant margin, of any algorithm available for the Atari ST.

Compressing free space eliminates free space from the area of used disk space by moving it all to the end or beginning of the disk. If there is any free space located within the allocated space on the active drive it will be eliminated. Compressing Free space does not defragment files, directory sectors, or directories it simply moves clusters so there is no free space within the allocated area. This basically segments your disk into two separate blocks, all of the used space is together and all of the free space is together.

So why would you want to use Compress Free Space rather than a Full Optimization? There are several reasons:

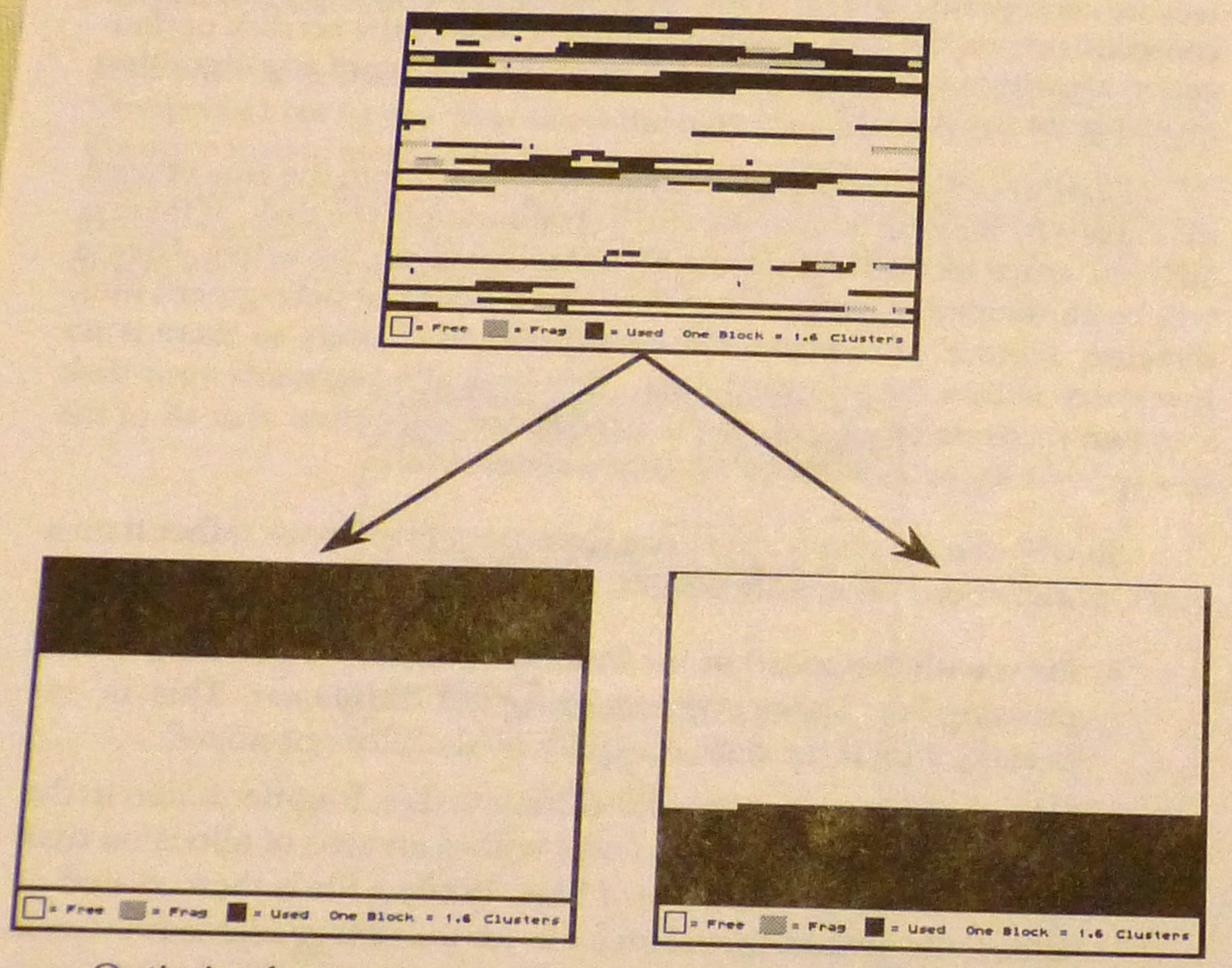
- Even with the speed of the Full Optimization algorithm, Compressing Free Space is generally a great deal faster. This is especially true if the disk was previously fully optimized.
- Compressing free space will eliminate disk fragmentation in the future. Free space accumulates within an area of allocated disk space through the deletion of files. When a file is then created or added, the operating system looks for the first available free space on your disk to place the new file. As soon as it finds some free space, it allocates the space to the new file. If all of your free space is consolidated in one block, then files added to your hard disk will not be fragmented.

✔ We recommend that all disks be fully optimized when you first receive Diamond Edge. After that you can use the compress free space option to keep the free space contiguous in between full optimizations. Use the disk fragmentation maps and statistics to assist you in determining the level of disk fragmentation on each of your disks.

Optimization Priority

Diamond Edge provides two optimization priorities: Optimization for **Reading** and **Writing**. A graphic description of the two methods is shown below.

Source Partition



Optimize for Reading

Optimize for Writing

You select the optimization priority by selecting the appropriate method under the Optimize menu. The basic distinction between the two priorities is where on the disk the allocated space is moved.

Selecting **Prioritize for Reading** will move all of the allocated disk space to the beginning of the partition. This improves the reading performance of the disk because all of the directory sectors and file clusters are physically closer to the FAT tables. This significantly reduces the amount of disk head movement that is required to find and then read a file. Disk head movement is the slowest component of any disk operation and the amount of disk head movement is critical to disk performance.

✔ In addition to the normal optimization function, when performing a read priority full optimization of Drive C all of the files contained in the Auto folder and root directory sectors are moved to the very start of the partition. The order of execution in the auto folder is not changed. This can dramatically speed up the boot process.

Selecting **Prioritize for Writing** will move all of the allocated disk space to the end of the partition. This improves the writing performance of the disk because all of the free space is physically closer to the FAT tables. The operating system does not have to work very hard to find the free space and when it does the disk head does not have to move very far to write the information.

✔ The optimum optimization method is a function of individual requirements. We highly recommend read optimization for your boot partition C. Beyond that, the choice is dependent on how you use your disks and personal preference. Experiment with both methods and use the method that gives you the best disk performance over time.

Other Optimization Options.

In addition to the Optimization Type and Priority, there are a number other options available to you when doing an optimization. The first three are: **Auto-Fix** errors, **Ignore Illegal File Names**, and **Save Lost Clusters**. These functions should be familiar and function exactly as described in the Disk Medic Chapter.

✔ Before each optimization, the disk is automatically subjected to a Disk Medic pass to assess it's ability to be optimized. If **ANY** disk errors are detected in the Disk Medic pass, they must be fixed before the optimization can proceed. Optimization of a disk that contains any errors is not allowed.

Remember to set the number of seconds to pause after a Disk Medic pass without errors, before Diamond Edge continues with the optimization. This is set in the Preferences Dialog. If no entry is found, then the program will stop and wait for user input before continuing with the optimization.

✔ Turning Visual update off will speed up the optimization of your drives by as much as 50%. Updating the fragmentation map takes a lot of time because drawing to the screen is very slow relative to disk accesses. If you do not need to watch the visual display and you care about saving time then turn the visual update off.

The other option involves the disposition of deleted files within the directories. A full description of what happens when you delete a file is given in the next chapter. The optimization of a disk drive significantly reduces the probability of successfully recovering a deleted file. If you select **Remove Deleted**, then the space occupied in the directory sectors for the deleted files will be removed. This will help speed up directory searches.

However, if you elect to keep the deleted files directory entries, and have a valid Disk Mirror file before the optimization, and the clusters containing the file you need to recover were not overwritten during the optimization, then there is a chance that you could recover some of the file. This could only happen if the file resided in the free space block after the optimization and the clusters were not used as swap space during the optimization. Not the best of odds, but still a chance.

Optimizing Your Drive

✔ Before you perform an optimization there is one thing left to do: **Backup your hard drive**. Diamond Edge provides secure protection against and recovery from a wide variety of disk related problems, however there is **NO SUBSTITUTE** for a valid up to date hard disk backup. Regardless of the safeguards within the program, the only way to protect against an unexpected power outage during an optimization is to have a current hard disk backup. We find that using Diamond Back 3 is the fastest and most reliable means to accomplish hard disk backups.

You may either optimize just the active drive by selecting **Optimize Drive** from the Optimize menu or on multiple drives by selecting **Optimize Multiple**. Before each disk is optimized, it is automatically examined by the Disk Medic for problems. Any problems detected by the Disk Medic must be fixed before optimization will be allowed. After the disk has passed the Disk Medic examination, the optimization will begin. A bell will ring to indicate that the optimization is complete.

6. Recovering Deleted Files

It happens to all of us sooner or later(some more frequently than others). You have just spent 3 hours working on an important paper and you decide to clean up your scratch directory. You start grabbing files and tossing them in the trash. Then suddenly your heart stops as you realize you just flushed the last three hours of work and all your ideas down the drain. Diamond Edge rescues you from this desperate situation by providing two different methods of recovering files that have previously been deleted including the ability to recover fragmented files.

What happens when you delete a file?

When you delete a file, all of the information regarding a file is not completely erased. In the directory sector holding the file, the first byte of the file name is replaced by the hexadecimal value 0xe5 (σ). This character tells the operating system that this file has been deleted and not to include it in directory searches anymore. So, the file READ.ME would become σEAD.ME. All of the rest of the directory information is left intact.

The next thing that happens is that all of the clusters allocated to the file in the FAT are cleared. This is the largest problem in recovering a deleted file. Although, the directory information remains intact, so we know where the file started, the FAT cluster links have been eliminated so we have no idea actually where the file is on the disk. If the file was not fragmented, then the clusters lie in consecutive order and recovery is relatively straightforward. However, if the file was fragmented full recovery is nearly impossible without knowledge of the FAT prior to deletion.

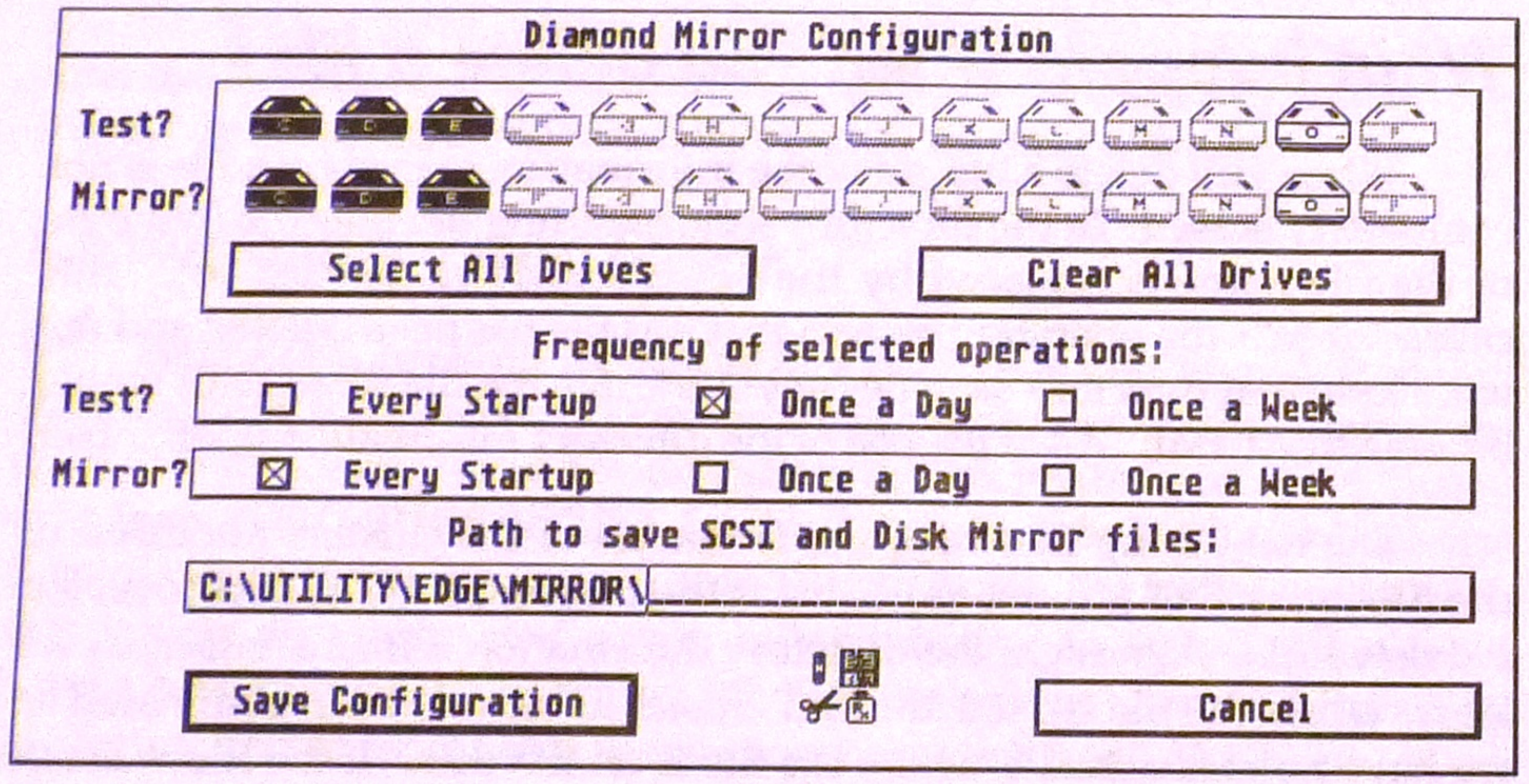
The good news is that nothing happens to the data on the disk. The data remains completely intact. The bad news is that now that the clusters have been released for use by the operating system, they can be overwritten at any time. Remember that clusters are allocated on a "first found, first used" basis. So, if your freshly deleted file happens to be the first available free space, the very next disk write will overwrite the data and make recovery impossible.

✔ The key to successfully recovering deleted files is 1) knowledge of the FAT prior to the deletion and 2) speed of action. The longer you wait to attempt to undelete a file the lower the probability of successful recovery. If you do accidently delete a file then do not perform ANY action that might write information to the disk before you use Diamond Edge to recover the file. Again speed of action is the key to successful file recovery.

Configuring Diamond Mirror

One of the keys to successful recovery of deleted files is knowledge of the FAT structure prior to deletion. Diamond Mirror is an auto folder program provided to save FAT and directory structure information at user specified intervals. It also has the capability of testing your disk structure at user specified intervals. Both of these functions are extremely critical and we ***HIGHLY*** recommend that you use Diamond Mirror.

Diamond Mirror is configured from within Diamond Edge by selecting **Configure DMIRROR** from the Undelete menu.



The following items can be specified:

- **What drive(s) to Test and Mirror:** Select the drives that you want Tested or Mirrored at boot time by clicking on the appropriate drive buttons. You may select as many or as few of your active devices. You may select different drives to test and mirror.
- **How often to Test and/or Mirror:** You can select to have Diamond Mirror test or mirror your drives every time your machine is booted up, once per day, or once per week. It is usually best to keep your Mirror files reasonably current. You can bypass any scheduled Mirroring by pressing ALT-S just before Diamond Mirror is scheduled to run or CONTROL-C while Diamond Mirror is running.
- **Where to put your Mirror files:** Specify the path that the Diamond Mirror files will be stored.

After you are satisfied with your configuration, select **Save Configuration**. If you want to exit without changing the existing configuration, select **Cancel**.

Choosing your Undelete method

Two methods of recovering deleted files are provided by Diamond Edge: **Simple Undelete** and **Mirror Undelete**. Select Mirror Undelete if you have a recent Mirror file for the drive that contains the deleted file. Otherwise, you should select Simple Undelete.

A **Simple Undelete** calculates the number of clusters the file would have occupied on the disk. Then it locates the starting cluster of the deleted file and calculates the number of consecutive clusters from the beginning cluster that are not currently allocated to other files. These unused clusters are then recovered.

A **Mirror Undelete** uses FAT and directory information stored in the disk Mirror files to determine exactly what clusters the file occupied prior to deletion The clusters that the file occupied, that are still not allocated to other files, are recovered. This is the only method that can fully recover a fragmented file. Both of these methods can assist you in recovering deleted files even if you regularly run DMIRROR.PRG.

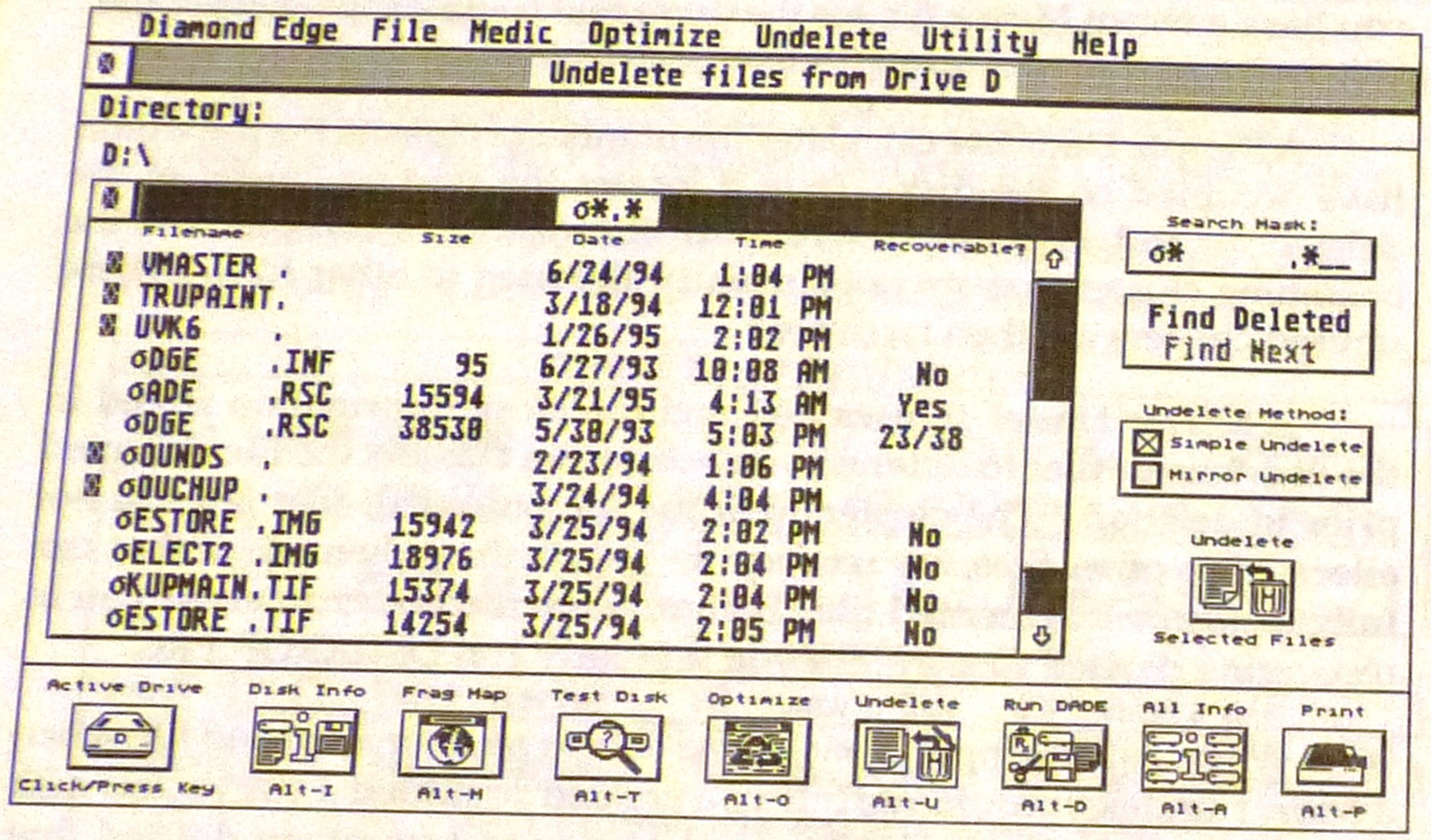
✔ When will a Simple Undelete be able to recover a deleted file when a Mirror undelete could not? If you created or added a file to you disk after the last time you ran Diamond Mirror and you then deleted that file, the Mirror file would contain no information about that file. In that case a Simple undelete would be your best chance at data recovery. Remember to keep your disks well optimized to give you the best chance of recovering deleted files.

Undeleting Files

When you select Undelete Files from the Undelete menu, the Undelete control screen is displayed. The Undelete File Selector functions exactly like the normal GEM file selector, except that only subdirectories and deleted files are shown. To open a subdirectory shown in the control screen double click on the entry. To move up to the parent directory again, click on the close box in the Undelete File Selector window. If you want to change the active undelete drive, then select the desired drive button from the available drive buttons.

✔ Each file that appears in the Undelete File Selector has an additional entry that indicates whether or not the file will be recoverable. If all of the clusters belonging to a file can be recovered then the recoverable entry is: Yes. If none of the clusters belonging to the deleted file can be recovered, then the recoverable entry is: NO.

However, if only some of the clusters belonging to the file are recoverable, then a ratio of the recoverable clusters to the total number of clusters will be shown. If the file in question is text then partial recovery is useful. Partial recovery of a binary or program file is seldom useful.



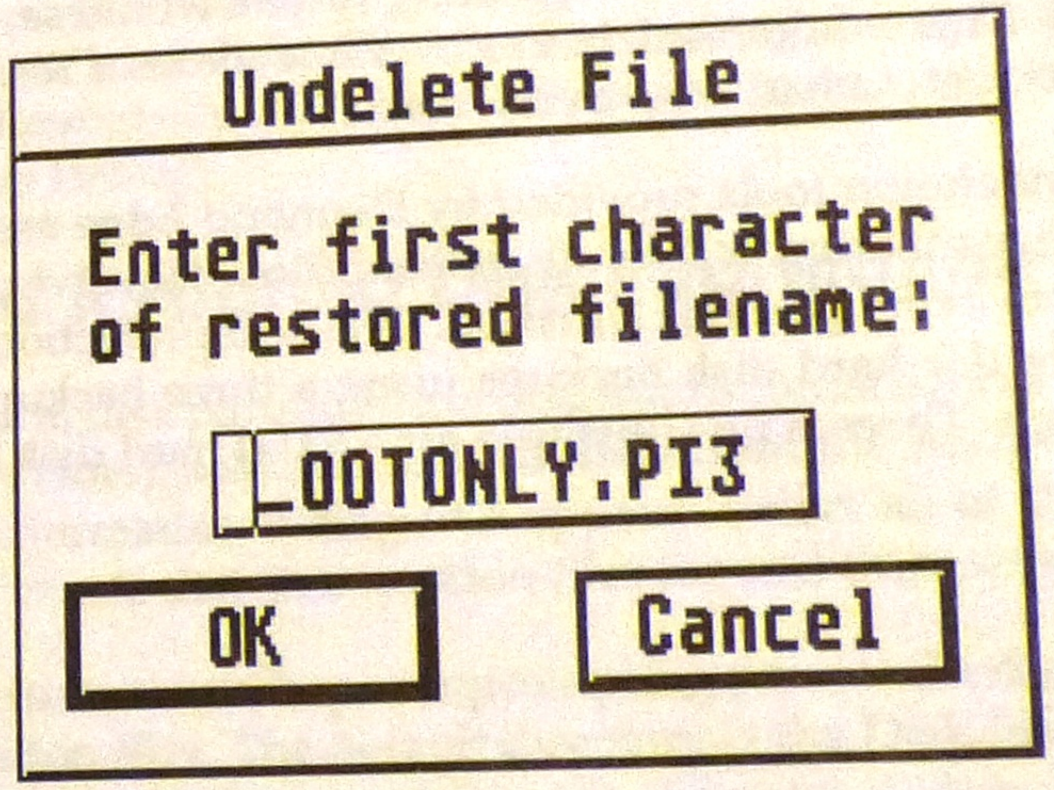
Diamond Edge provides a general purpose deleted file finder within the Undelete control screen. This is especially useful if you have forgotten the name of the file(s) you want to recover or simply don't remember what directory the file used to reside. To search for deleted files enter a deleted file search mask in the editable field in the Undelete Control screen labeled Search Mask. You can search for a specific file or enter a general purpose wildcard to find all files that match the wildcard criteria.

The two standard wild cards on the Atari ST are * and ?. The * appearing in a wild card search mask will backup all files containing a character string in the *'s position. The ? appearing in a wild card search mask will backup all files containing a single character in the ?'s position.

For example, if you wanted to search for all deleted files with a .PI1 extension., you would enter s*.PI1 for the Undelete Search Mask. If you had wanted to search for all deleted files with extension of PI1, PI2, or PI3 you would enter s*.PI? as the Undelete Search Mask.

To initiate the deleted file search, select **Find** from the Undelete Control Screen. You can also press the return key to initiate deleted file search. The program will search the current drive for the first file that matches the wildcard specification. If a match is located, the Undelete File Selector will automatically move to the directory containing the file and select the file. If you want to find additional deleted files that match the current wildcard mask, select **Find Next** from the Undelete Control screen.

✔ To select a file to undelete, click on the file name in the Undelete File Selector. When the entry appears **selected** it is marked for undeletion. You may select as many files as you want to undelete in the same directory by holding the shift key down while selecting the file. After you have selected all of the files that you want to undelete in the current directory, then select **Undelete** from the Undelete control screen. For each file that you select you will be prompted to enter the first character of the deleted filename.



The program then checks to see if there is another file in the directory that already has that name. If a conflict is found, you will be prompted to enter a different first character.

✔ In order to undelete files from a subdirectory that has also been deleted, you must first undelete the directory. When you have finished undeleting all of the files that you require, select **Cancel** to return to the Diamond Edge basic window.

What to do after you undelete a file

After you have undeleted all of the files that you require, you should verify the validity of the recovered files. If you have a current validation file for the drive, then perform a validation pass to assess the state of the undeleted files. This will tell you if the undeleted file is the same as before or if the files data has been corrupted. If you do not have a current validation file, then you should manually examine the files you just undeleted to determine if they are valid.

Verification of the undeleted file is required because even with Mirror protection, the validity of the file recovery is not 100% assured. If you deleted some files and then add other files, the "first available first used principal" would apply and the new files would overwrite the area of disk that the deleted file occupied. Then if you deleted those new files before the next time that you run Mirror, there would be no record of that activity. The Mirror file would tell you where the originally undeleted file resided on the disk, and the operating system would say that space is unallocated. This would result in a recoverable status of Yes, even though the files information had been overwritten.

The undeletion tools provided by Diamond Edge are designed to assist you in recovering accidently deleted files. However, they cannot save you from every possible situation. The best protection against loss of data is regular hard disk backups using a three backup set generational strategy. There is no substitute for a valid hard disk backup.

7. Archiving Disk Information

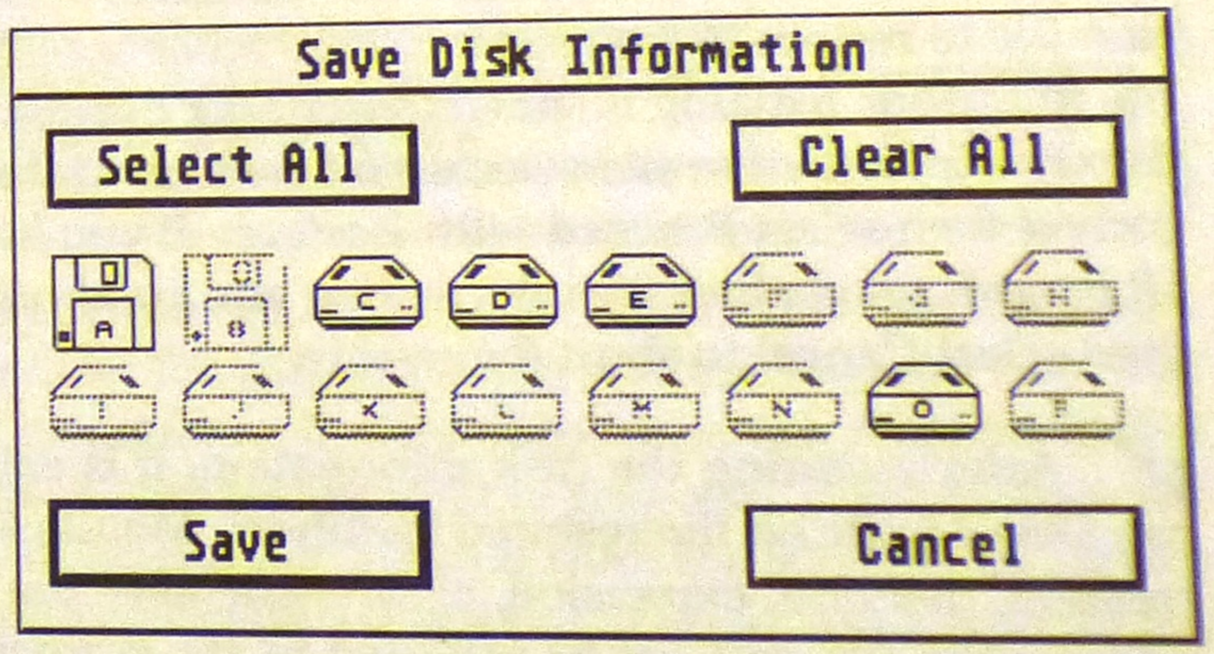
Catastrophic failures of the disk operating system, the hard disk driver software, or applications programs occasionally cause the total loss of critical disk information. The most often effected areas include the partitioning information located in physical sector 0 of the hard disk, the partition boot sector, the partition File Allocation Table, and the partition root directory sectors. The data on your hard disk is often totally unaffected, however you can't access ANY of it because the information that tells the operating system how to find it is either gone or corrupted.

✔ While there is still no substitute for a valid hard disk backup, the easiest and fastest way to recover from these situations is to simply reinstall a copy of the corrupted information. We **HIGHLY** recommend that you keep a current archive of disk information on all of your hard drive partitions and current SCSI partitioning information on all of your SCSI drives. We also recommend that you keep this data on a floppy disk in addition to on your hard disk. The creation and regular updating of a Diamond Edge Emergency disk described earlier is an excellent catastrophe prevention strategy.

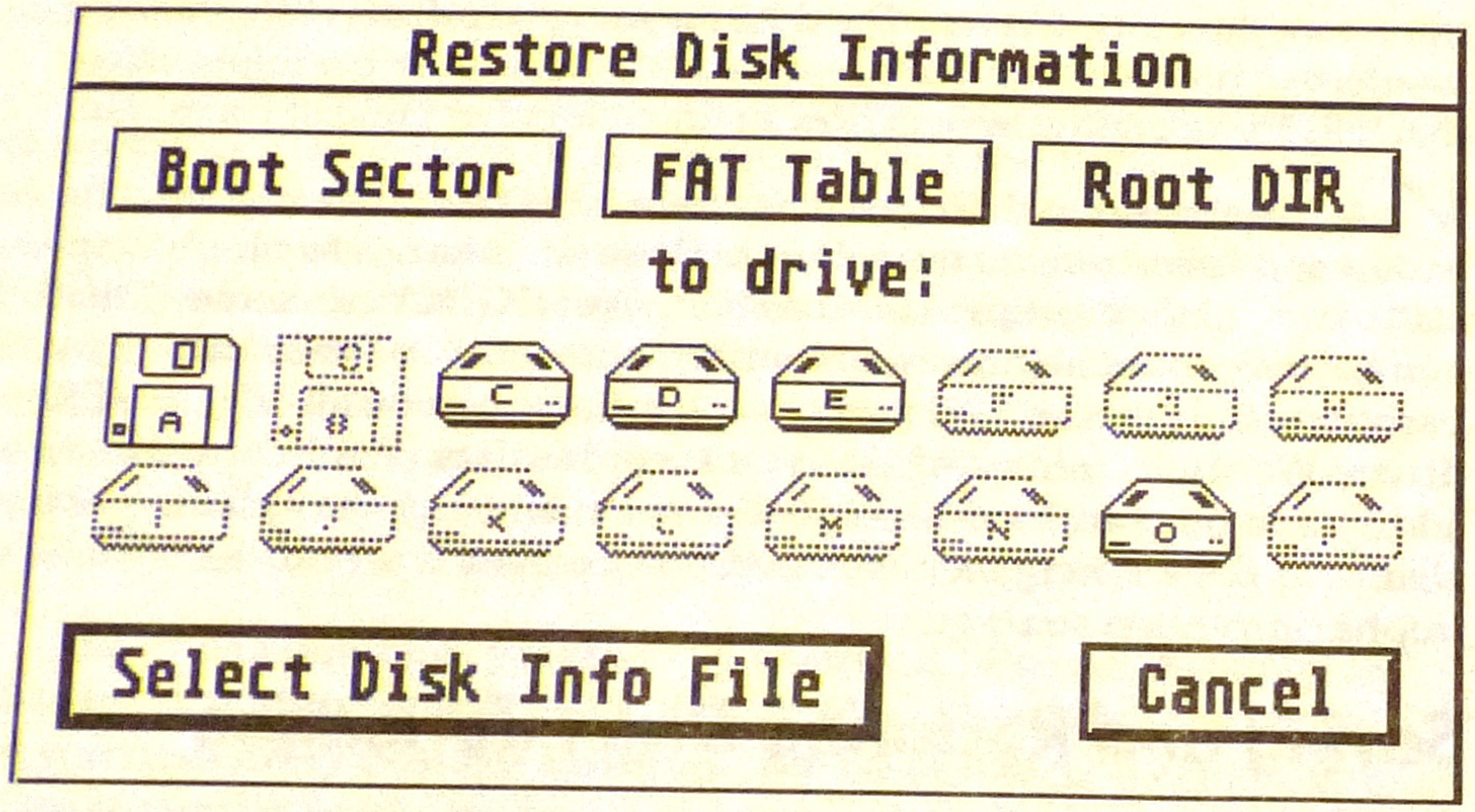
Saving and Restoring Disk Information

To save critical disk partition structure information(boot sector, FAT table, directory sectors), select **Save Disk Info** from the Archive menu. The Save Disk Information dialog will appear. Select all of the drives that you want to create disk information files for and then select **Save**.

A GEM file selector will appear prompting you for the directory to save the information files. The default directory is the Disk Information directory specified in the Diamond Edge Preferences dialog. Only the destination path is required, the disk information file names are automatically generated. Select **Ok** from the File Selector to save the disk information files. Select **Cancel** to abort the process.



To restore the critical disk information to a disk, select Restore Disk Info from the Archive menu. The restore disk information dialog will appear. You have the option of restoring the boot sector, the FAT table, the root directory sector, or any combination. Then choose the drive that you want the information restored. It is often advisable to first reinstall the boot sector, perform a Disk Medic check with Auto-Fix **OFF**. If significant problems remain, then reinstall the FAT table and root directory sectors.



✔ If the boot sector is severely damaged, TOS may refuse to recognize the disk as valid. It may not be listed as active or may not be recognized at all(i.e. one less drive appears than you think you have). If this occurs, it will be necessary to rebuild the partition from the Hard Disk Partitioning function described earlier, before you can reinstall the archived disk information.

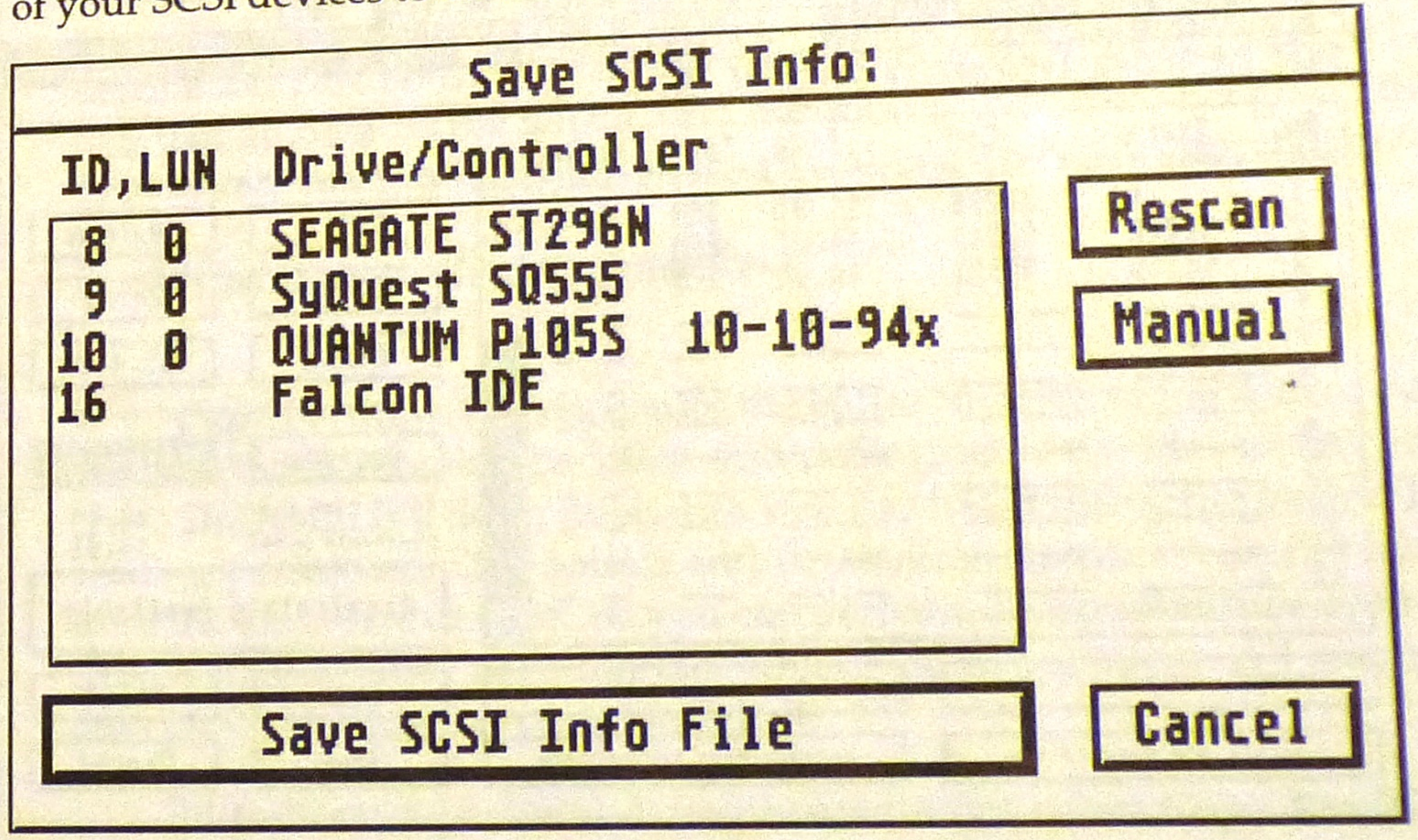
Select the **Select Disk Info File** button to choose the disk information file to restore to the selected disk. After specifying the information file to restore, a dialog is shown describing the disk characteristics from the disk information file selected for restore. If the disk information is correct then select **Proceed with Restore**. If you have **any questions** at all regarding whether the information you are about to restore is correct, then select **Cancel** to abort the restore.

✔ After restoring the disk information, it is extremely advisable to run Disk Medic on the restored partition. Modifications to the disk(file creation, deletion, expansion, etc.) made after the creation of the disk information file will not be reflected in the restored information. This could lead to some loss of data created or modified after the disk information file was created. Much of this information can be recovered through the undelete function or from the Lost cluster save file produced by the Disk Medic. There is no substitute for an up to date hard disk backup or an up to date disk information file.

Saving and Restoring SCSI Partition Info

The total loss of the hard disk partitioning information located in physical sector 0 of your hard disk is more common than most people believe. This is an easily recoverable situation with absolutely no loss of data, if you have a SCSI information file saved. However, if you don't have the SCSI information saved, only the most skilled and knowledgeable hackers can resurrect your disk.

To create a SCSI information file, select Save SCSI Info from the Archive menu. Diamond Edge will scan ST ACSI identification numbers 0-7 and TT SCSI or Falcon SCSI-2 identification numbers 8-15 and Falcon IDE id 16 for active devices. The drive identification of the devices found will be displayed in the Select SCSI Device dialog. Select all of your SCSI devices to create information files for each SCSI drive.

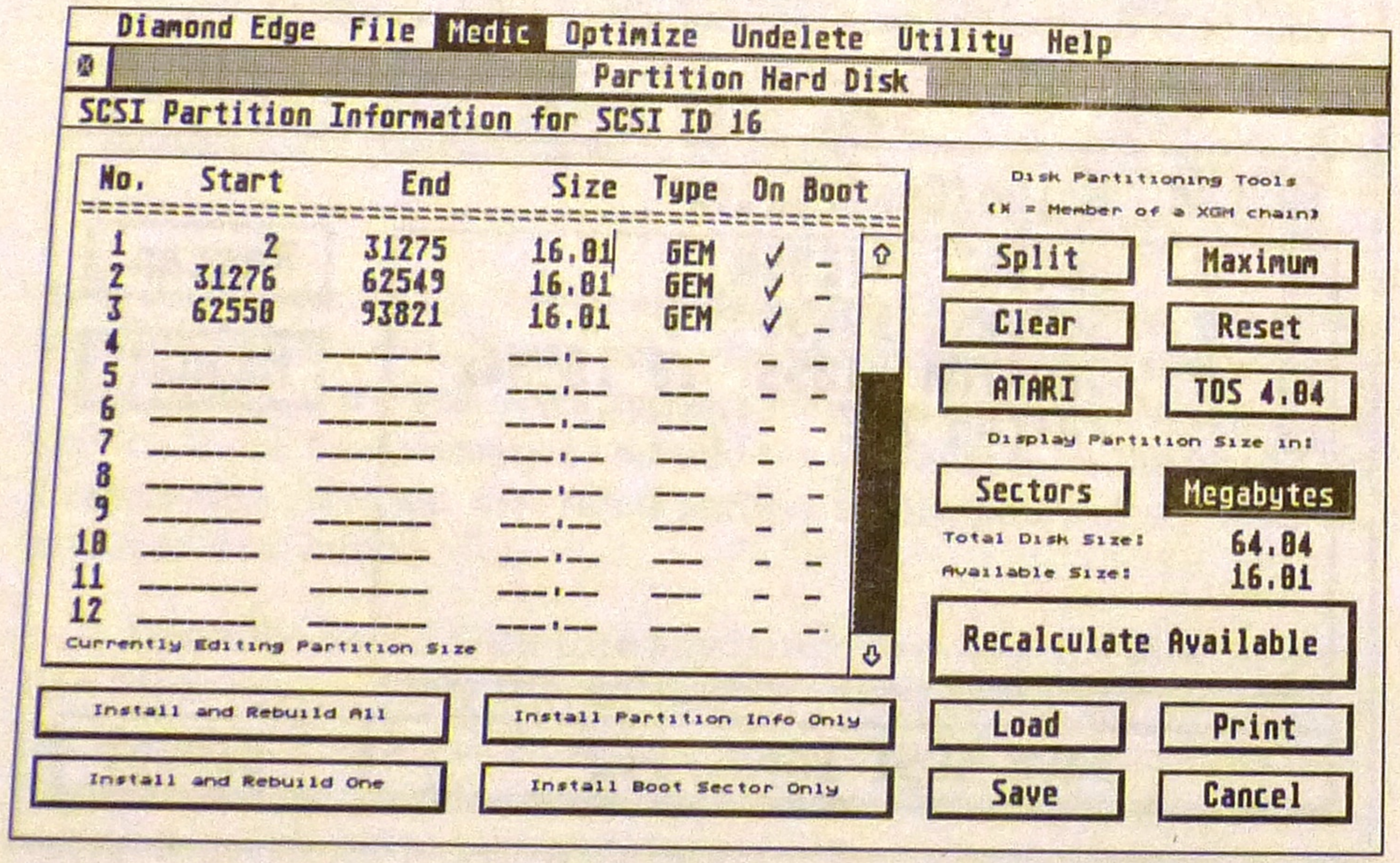


Some early SCSI drives and MFM drives attached to the ST using a SCSI-> MFM adapter(like the Adaptec 4000) will not respond correctly to the SCSI Inquiry command. If your drive does not respond to the inquiry select **Rescan**, it may just be slow to respond. If it still does not respond to the inquiry, then select **Manual** from the Select SCSI Device dialog. A manual selection dialog will appear to allow you to specify the SCSI ID Number of the device that you want to partition.

Selecting Save SCSI Info File will bring up a GEM file selector prompting you to enter the directory path to save the SCSI information files. The default directory is the Disk Information directory specified in the Diamond Edge Preferences dialog. Only the destination path is required, the disk information file names are automatically generated. Select **Ok** from the File Selector to save the SCSI information files. Select **Cancel** to abort the process.

To restore the disk partitioning information to a SCSI disk, select **Restore SCSI Info** from the Archive menu. Select the SCSI drive that you want the information restored and then press **Select SCSI Info File.** A GEM file selector will appear allowing you to select the SCSI information file to restore to the selected drive. Select **Ok** to proceed with the SCSI information restore.

The standard Hard Disk partitioning screen will now appear with the loaded with the partitioning information contained in the file. To reinstall the old partitioing information without effecting any other data on the disk, select "Install Partition Info Only".



If you have **any questions** at all regarding whether the information you are about to restore is correct, then select **Cancel** to abort the restore

8. General Utilities

Diamond Edge provides a number of general disk maintenance utilities to help you get the most out of your disks.

Partition Copying

Diamond Edge provides two general purpose disk to disk copying methods for different applications.

Partition Copy - Image

Copy Partition Image copies sector for sector an exact image of the source partition to the destination partition. The directory structure and disk fragmentation characteristics of the source are exactly duplicated on the destination. The source and destination partitions must be of exactly the same size and type(both GEM or both BGM). All existing data on the destination partition is lost.

To perform a partition image copy, select **Copy Partition Image** from the Utility menu. You will be prompted to enter the source and destination partitions. If you have **Warn** mode set to then you will be told that all data on the destination partition will be lost and given the opportunity to cancel. Otherwise the partition copy begins immediately.

Partition Copy - Defragmented

✔ **Copy Partition Defrag** copies the entire contents of the source partition to the destination partition and defragments the files in the process. The result is a destination disk that has had a full optimization for reading performed. The source and destination partitions do not have to be the same size or type. So you can perform a Copy Partition Defrag from a 10 Meg GEM partition to a 40 meg BGM partition. All existing data on the destination partition is lost.

To perform a partition defragmented copy, select **Copy Partition Defrag** from the Utility menu. You will be prompted to enter the source and destination partitions. If you have **Warn** mode set then you will be told that all data on the destination partition will be lost and given the opportunity to cancel. Otherwise the partition copy begins immediately.

Forcing Media Change

TOS occasionally has difficulty detecting whether a disk has changed state and requires updating it's internal information about the disk. This is especially true regarding floppy disks. If you suspect that you have modified your disk and TOS does not recognize the change, then you can choose the disk as the active disk and select **Force Media Change**. This will force all versions of TOS to flush and rebuild all information it has about the disk.

Browsing, Viewing, and Printing Files

These functions allow you to look around your disk, look at text files, and print text files. Selecting **Browse** will give you access to the GEM File Selector to look around you disk. Selecting **View File** allows you to view a text file on the screen. Selecting **Print File** will allow you to print a text file of your choice on your printer. This is useful to examining and printing Disk Medic Error reports or Validation Error reports.

Partition Zeroing

If your disk structure has become unrecoverably corrupted or if you just want to easily erase all of the files on your disk, Diamond Edge provides several methods of Zeroing your disk. Zeroing a disk means erasing all of the information in the disks FAT and root directory sectors. The result will be a completely empty partition.

Because zeroing a disk is not something you want to do accidently, you must enable the zeroing functions by selecting **Enable Zeroing** from the Utility Menu.

Zero Partition

To Zero a partition, make the partition you want to zero the active partition. Then select **Zero Partition** from the Utility menu. Regardless of the warning level specified in the user preferences, you will always be warned before a zeroing operation. If you decide to continue with the zeroing select **Yes**, if you want to abort the zeroing select **No**.

✔ After you have confirmed your intent to zero the partition, but before the zeroing is actually performed all of the critical disk information is saved in a disk information file. This will allow you to Unzero the partition with absolutely no loss of data if you decide that you have mistakenly zeroed the partition.

Unzero Partition

To Unzero a partition, make the partition you want to Unzero the active partition. Then select **Unzero Partition** from the Utility menu. The same procedure is followed to unzero a partition, as to restore the disk partition information.

After selecting the disk you want to unzero, select the **Select Disk Info File** button to choose the disk information file to restore to the selected disk. After specifying the information file to restore, a dialog is shown describing the disk characteristics form the disk information file selected for restore. If the disk information is correct then select **Proceed with Restore**. If you have **any questions** at all regarding whether the information you are about to restore is correct, then select **Cancel** to abort the restore.

✔ After, unzeroing the disk it is extremely advisable to run Disk Medic and File Validation on the unzeroed partition. Any modification to the disk(file creation, deletion, expansion, etc.) made after the disk was zeroed can cause corrupted disk structure and information on the unzeroed partition. If you unzero a disk immediately after zeroing it, the recovery will be complete with no loss of data. If you have written anything to the disk and then unzero it you will have at least some corrupted information.

Wipe Partition

Wiping a partition is a special kind of Zeroing. When you wipe a partition, not only are the FAT tables and root directory sectors erased, but every sector of the partition is overwritten. This is especially useful if you need to completely eliminate highly confidential data.

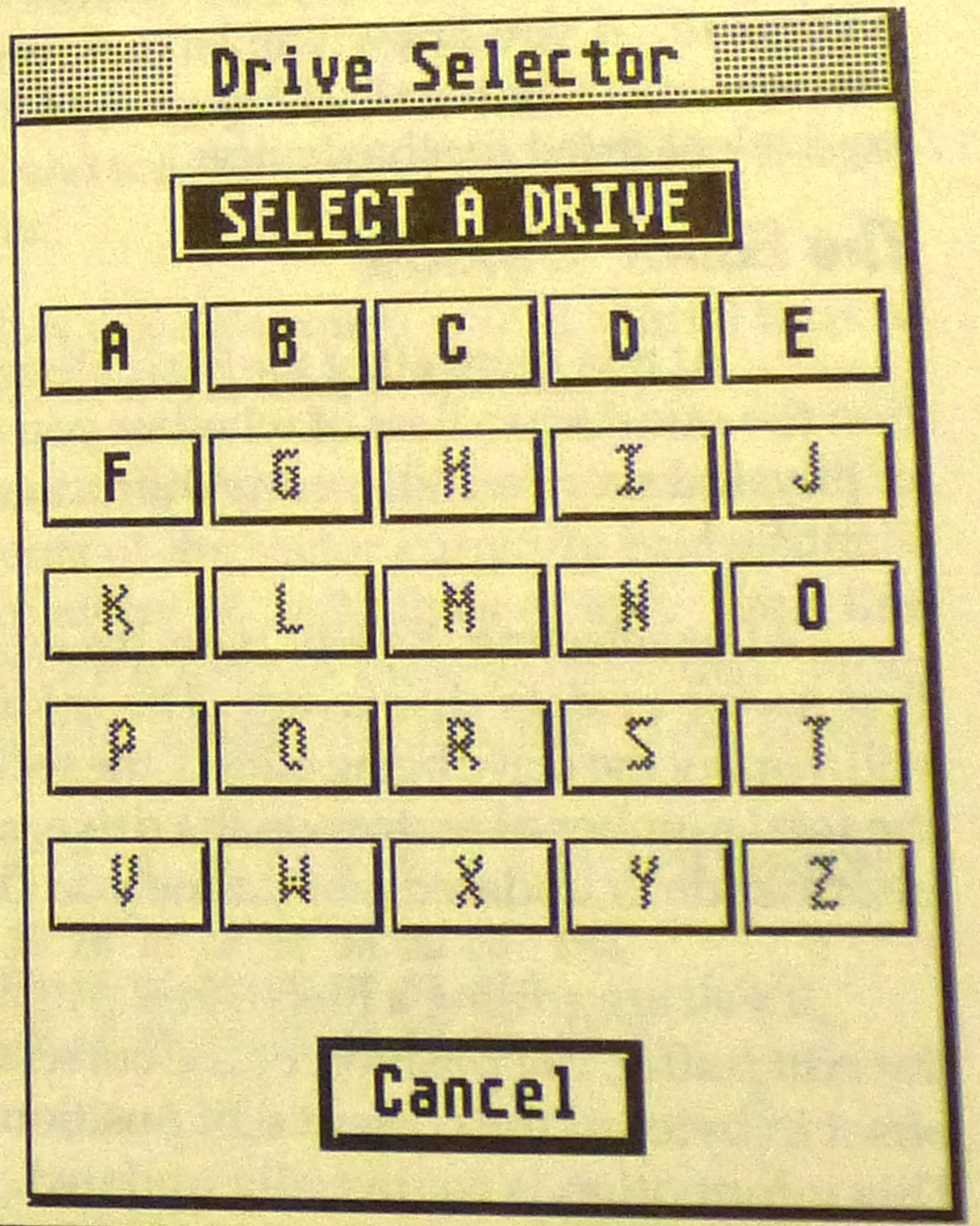
9. Diamond Advanced Disk Editor

The Diamond Advanced Disk Editor(DADE) is a powerful sector editor. It offers powerful features for advanced users while still being easy to use for novices. DADE is designed to edit files, logical sectors(sectors on a mounted logical drive partition like C: etc), and physical sectors(the actual sectors on a physical SCSI device). It can accommodate logical sectors from 512 bytes up to 32K in size, and physical drives of any size.

✔ Extreme caution should be exercised when using DADE. While it has incredible power to fix absolutely any disk problem or recover any file, the power comes at a price. It is just as easy to destroy your disk with DADE as is to fix it, if you attempt to perform functions without knowing what you are doing. If you have any doubts whatsoever, then read the manual again. If you still have doubts, then contact technical support.

Using DADE

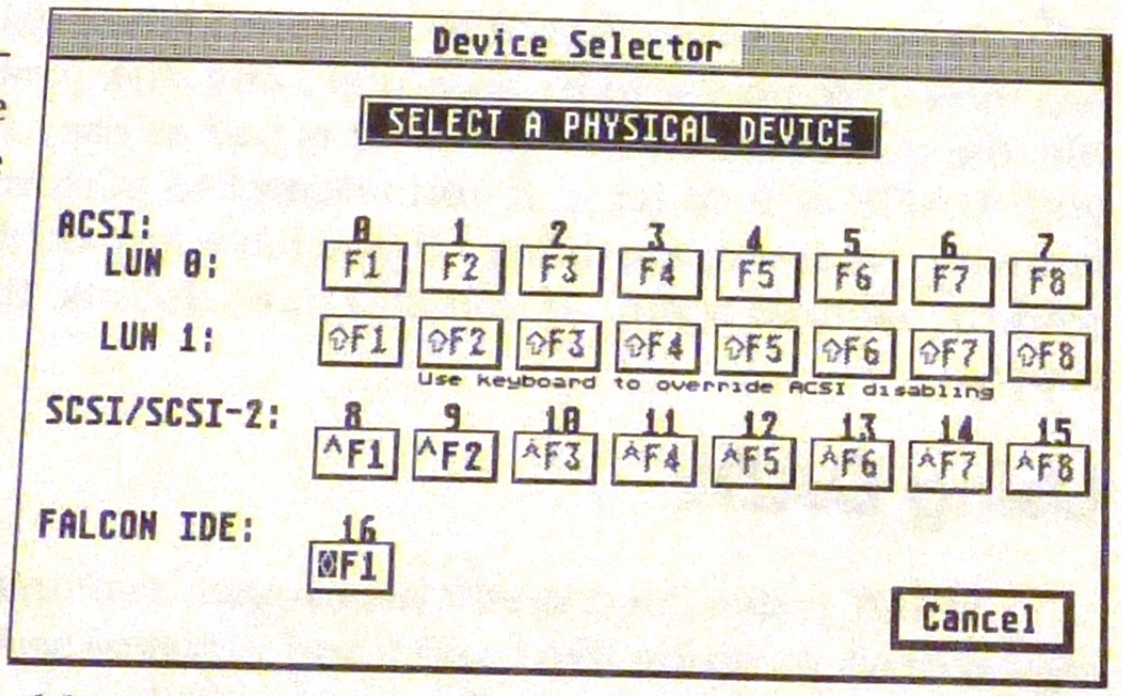
DADE is designed to edit files, logical sectors(sectors on a mounted logical drive partition like C: etc), and physical sectors(the actual sectors on a physical SCSI device). To start an editing session, select the type of editing you would like to do from the File menu. If you want to edit a file, then a GEM file selector will appear allowing you to choose the file to edit. If you want to edit a sector on a logical partition of your drive, then the Drive Selector dialog will appear.



Partitions that are not available for editing will appear disabled. To select a drive to edit, place the mouse pointer on the appropriate drive letter and click. The editor window will then open and sector 0 will be displayed. If you decide not to edit a drive, then select Cancel.

If you want to edit sectors at the physical device level, then select Edit Device. This capability is especially useful in repairing very badly damaged drives. For example, say the boot sector on one of your partitions is damaged. Your hard disk driver will not recognize the partition so you can't use the normal tools in Diamond Edge or the logical editing mode of DADE. Editing the drive at the physical device level is the only way to repair the damage and hope to recover the rest of the drive.

First DADE will scan the ACSI and SCSI busses for active devices and then will display the Select Device dialog. Devices that are not active in the system will be disabled.



To select a device to edit point the mouse pointer at the appropriate drive button and click the left button. The editor window will then open with physical sector 0 of the device selected displayed.

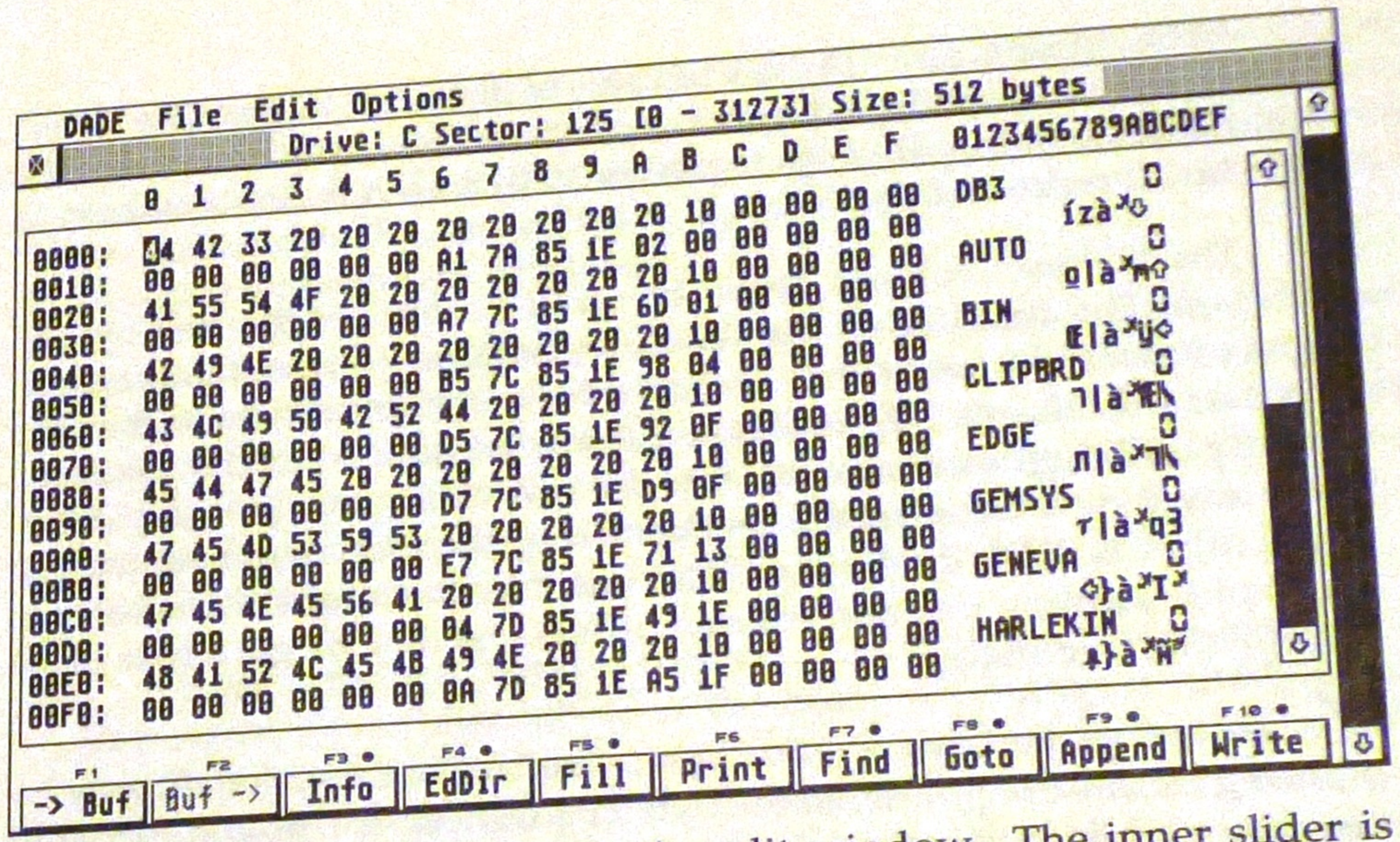
Note that some older SCSI drives may not respond to the Inquiry command. If you know you have a device at a particular ID number, you can override the disabled state of a button by using the keyboard equivalent listed for the device.

The Editor Window

At this point all of the basic elements of the editor window function the same regardless of whether you are editing files, logical sectors, or physical sectors. Where any differences exist, they will be specifically identified.

After selecting the edit type, the editor window will open with the first sector of data displayed. The information line of editor window will display the drive being edited, the sector number of the current buffer, the total number of sectors on the drive, and the size of each sector. This information is updated every time you change sectors.

If you are editing a file, then it will display the file name, the size of the edit buffer, the position of the current edit buffer in the file, and the offset in bytes of the current edit position from the beginning of the file. This information is continually updated.

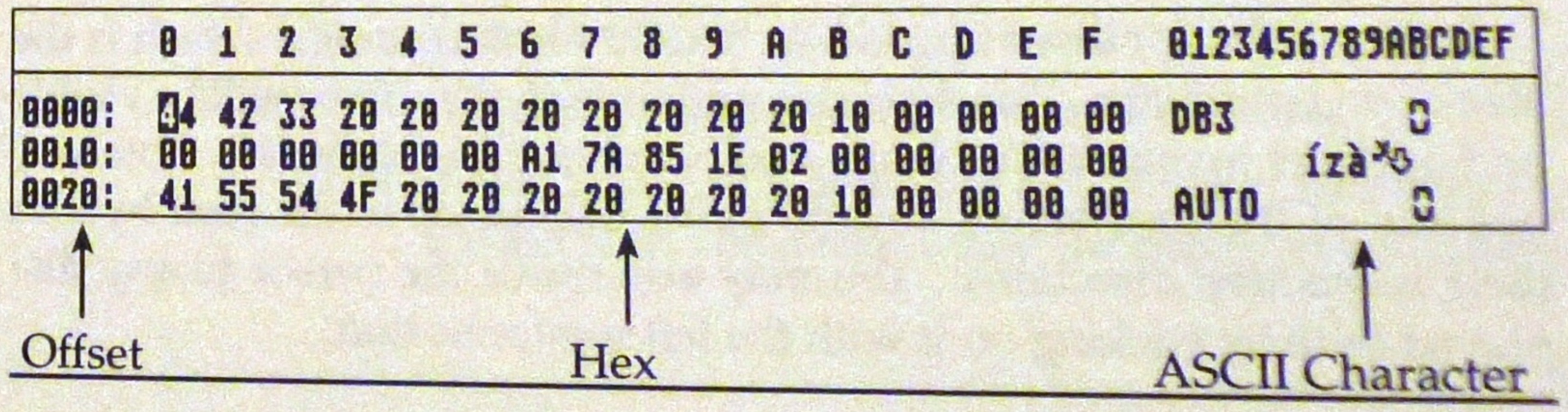


You will see two sliders in the edit window. The inner slider is active if one sector of data is too large to be displayed on a single screen. Scrolling with the inner slider moves you within the current sector. The outer slider is used to scroll through the different sectors of the logical or physical disk.

When editing files, the buffer size per screen is inherited from the logical sector size of the drive the file resides on. For example, if drive D has a logical sector size of 2K, then any file you edit on drive D will also have a logical sector size of 2K. Scrolling with the inner slider moves you within the current 2K buffer and scrolling with the outer slider moves you between 2K blocks of the file.

At the bottom of the window is a convenient row of special feature buttons. These will be explained in detail later in this chapter.

Inside the window, enclosed in a box known as the sector editing work area, you'll find the contents of the sector currently being edited. In this editing area you will see either 16 or 32 lines of text. Each line consists of three basic elements (we'll refer to these as columns). From left to right the three columns are the "offset" column, the "Hex" column and the "Character" column.



Each editable line displays 16 bytes of the sector buffer. The "Hex" column displays the 16 bytes as two digit hexadecimal numbers. The "Character" column displays the same 16 bytes but it displays them as their ASCII representation.

The "Offset" column gives the offset (in hex) of the first byte in each row. The first line of the window contains hexadecimal(base 16) digits numbered from '0' to 'F' over the "Hex" column and also over the "Character" column. These Hex digits are the offset of each byte in the row. By adding the column offset to the row offset you can easily determine any byte's offset in the sector. Using the picture above, the last row's offset is 01F0, the last hex byte (last two hex digits) has a column offset of F. Adding row and column offsets together we get a byte offset of 01FF (511 decimal).

✔ Note that you can get the byte offset within a sector directly by pointing the mouse pointer at the byte in questions and holding the shift key down while you click and hold the left mouse button. A pop-up box will display the offset of that byte from the start of the sector.

Editing a sector

When editing sectors it is always important to remember about how DADE manages the data you are editing. DADE maintains two separate buffers. The first of these buffers is known as the "editing buffer", the second, as the "copy buffer".

✔ Whenever you instruct DADE to display a sector, DADE reads the sector, placing a copy of it into the "editing buffer". The contents of the "editing buffer" are displayed in the editing area at all times. You may freely edit the buffer and some of DADE's options will also change the contents of the buffer. However, for these changes to be permanent you must SAVE the sector back to disk or file using the Write button.

The "copy buffer" is a sector sized buffer used for block copy operations. It contains data from the last CUT or COPY operation that was performed on a marked block. You can change drives or files you are editing without affecting the contents of the copy buffer.

The editing cursor appears as a solid black rectangle which is the size of one character. The character underneath the cursor will be visible and appears inverted (i.e.: white text on a black background). You may use any of the four arrow keys to move the cursor to the next byte in their respective directions. You may also move the cursor to any displayed byte by clicking on it with the left mouse button.

Using the arrow keys, the cursor will remain inside of the current column ("Hex" or "Character".) Using the mouse, you can position the cursor in either column by clicking on the byte in that column. Moving the cursor with the arrow keys it is possible to view the entire sector if it can not be completely displayed.

As stated earlier, the "Hex" column displays a number of rows, each of which displays 16 bytes. Each byte is displayed in it's hexadecimal equivalent. That is, each byte is displayed as a set of two hexadecimal digits (0 - 9, and A - F). The first hexadecimal digit of a byte we shall call the high nibble. The second will be referred to as the low nibble. Each nibble represents 4 bits of a byte (8 bits).

When editing in the Hex column, DADE allows you to position the edit cursor over either nibble of the byte. To change a nibble's value simply position the edit cursor over the desired nibble and type in the nibble's new value. The edit cursor will automatically advance to the next nibble. DADE does not force you to modify the whole byte. Any change made to either nibble of a byte will be reflected in that byte's character in the "Character" column.

When editing in the "Character" column the character under the edit cursor will be replaced with the ASCII value of the key pressed. Any change will be reflected in that byte's hexadecimal value in the "Hex" column.

Most CONTROL + <key> combinations will produce the expected character. The exceptions to this are the control key combinations which serve as menu equivalents. Holding the ALTERNATE key gives you access to some extended ASCII characters. By far the easiest way to enter control and extended characters is by editing their bytes in the "Hex" column. Then you just enter the characters hexadecimal equivalent.

Special Editing Keys

When the sector editing window is topped several keys have a special significance:

TAB	Toggles cursor from one editing column to the other.
UNDO	Rereads current sector into editing buffer.
HELP	Toggles cursor between solid and blinking.
	(Note: DADE will default to a blinking cursor if you hold the left shift key down while the program is loading.)

Block Editing

DADE allows you to perform block editing functions. To mark a block for editing, position the editing cursor over the starting byte of the block and select *Block Start* from the Edit Menu or press CONTROL-S. Then place the editing cursor on the last byte of the desired block and select *Block End* or press CONTROL-E. The bytes contained within the marked block will then be shown as selected.

The following functions may be performed on the block:

Cut Block — Copies selected block into the copy buffer and replaces the block with zeros. CONTROL-X

Copy Block — Copies selected block into the copy buffer and leaves the source block unchanged. CONTROL-C

Once you have the selected block in the copy buffer, there are several actions that you may now take:

Paste — You may paste the contents of the copy buffer by selecting an insertion point with the mouse and then selecting Paste from the edit menu or by pressing CONTROL-V. The contents of the copy buffer will be copied to current buffer starting with the byte indicated by the edit cursor.

✔ Note that once you have copied data into the copy buffer, you can change sectors, drives, or files and then paste the copy buffer data into the new current buffer.

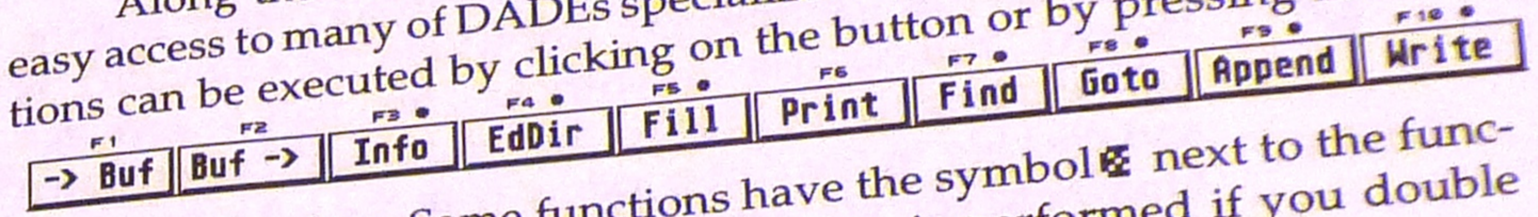
Save — A GEM file selector will appear allowing you to save the contents of the copy buffer to a specified file. The file does not need to exist. If it does then it will be overwritten by the information in the copy buffer.

Append — A GEM file selector will appear allowing you to append the contents of the copy buffer to a specified file.

To unmark a block select Unmark Block from the edit menu or by pressing CONTROL-U.

Editor Button Bar

Along the bottom of the editor window is a button bar allowing easy access to many of DADEs specialized functions. Each of these functions can be executed by clicking on the button or by pressing the indi-



cated function key. Some functions have the symbol ⊠ next to the function key. This means that a special action is performed if you double click on that function. These are described in detail below.

-> Buf, Buf ->

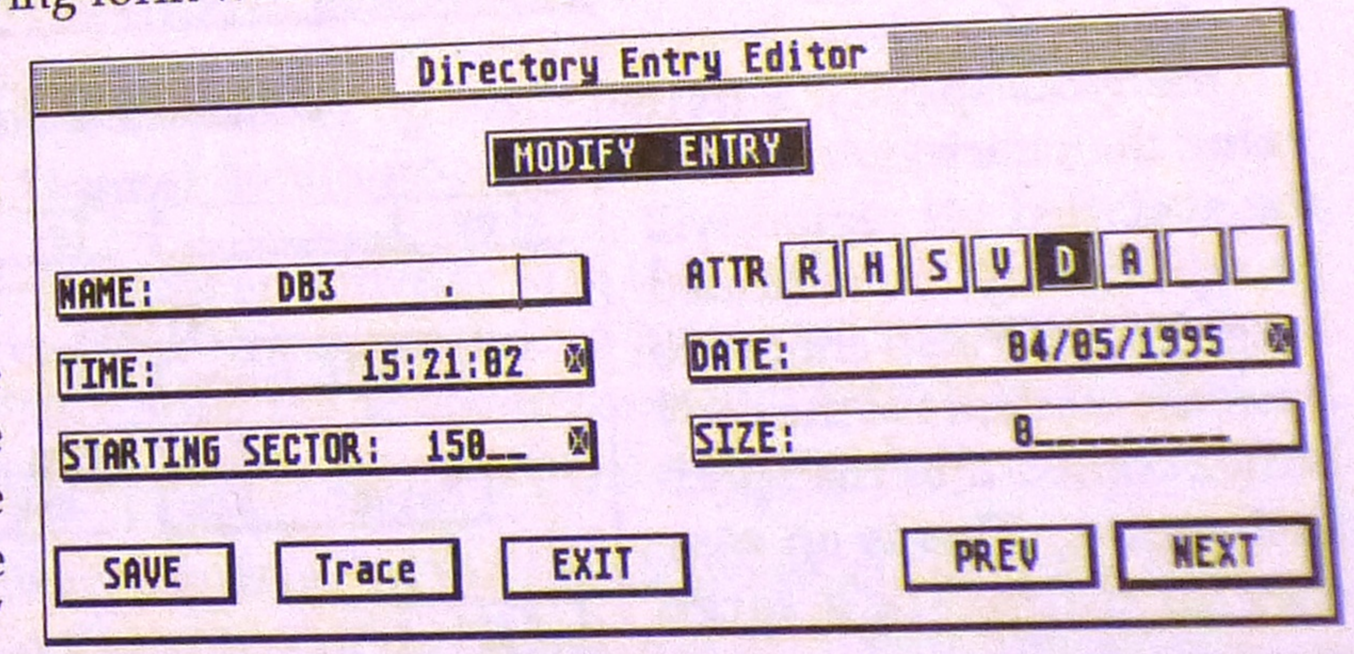
This function copies the entire current edit buffer into the copy buffer, or the copy buffer into the current editing buffer. Note that these two functions differ from Block copy and paste operations in that they operate on an entire sector buffer at a time.

Info

Selecting this button will display important disk statistics similar to those shown on the Disk Information screen in the main Diamond Edge program. Double clicking on this function will bring up extended information about the sector currently in the edit buffer such as if it belongs to a critical disk structure or the FAT chain if the sector belongs to a file on the disk.

EdDir

This function brings up a form that makes editing and correcting directory entries easy. To execute the function place the editing cursor on the directory entry that you want to edit and click on the EdDir button. The following form will then be displayed:



The directory entry is displayed in a decoded form for easy editing. Double clicking on the date or time will enter the current date/time into that field. Double clicking on the starting sector will jump you to that sector and load it into the edit buffer. This is a very convenient method of quickly traversing a directory tree.

There are a number of different actions that you can take when using this form:

Save Copies your modifications to the directory entry into the edit buffer in the directory slot where the edit cursor is located. You can use one entry as a template to save in a different location by holding the right mouse button down and clicking in the edit window with the left mouse button where you want the edit cursor to be placed and then select Save to save to the new location.

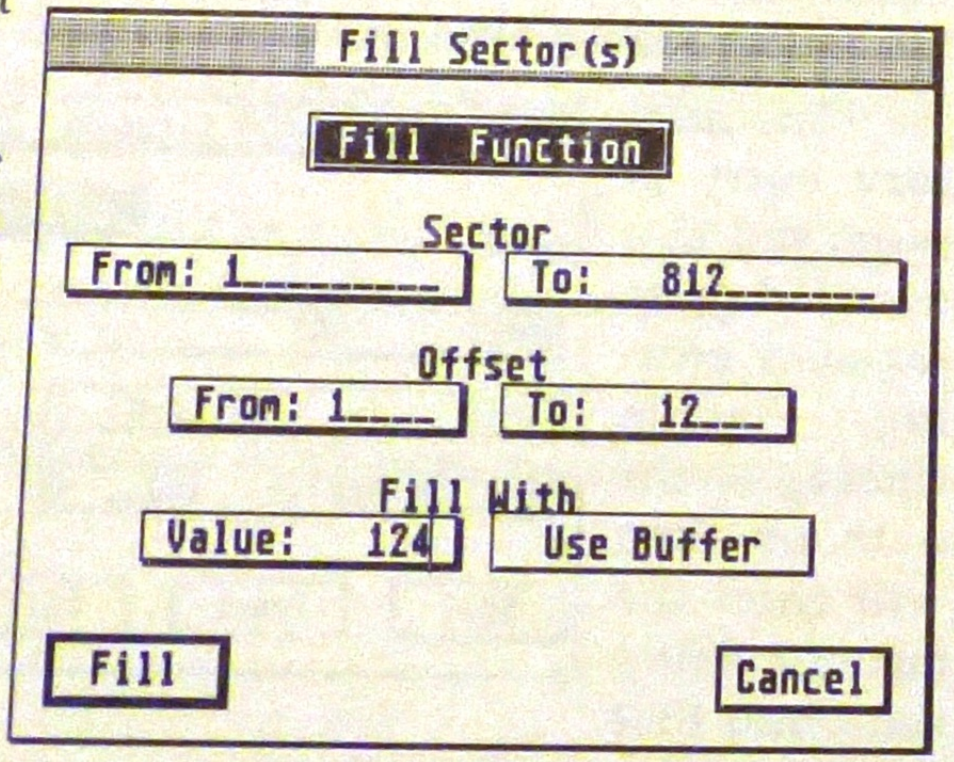
Trace This function displays a trace of the FAT chain of a directory. Very useful for manual recovery. This function can also be invoked without entering the Edit Directory form by double clicking on the EdDir button.

Exit This button will exit the Edit Directory form without making any modifications to the edit buffer.

Prev, Next These functions automatically move the Edit Directory form to the Next or Previous Directory entry in the current edit buffer without leaving the form. This allows you to edit many directory entries without repeatedly entering and leaving the edit form. Remember to Save changes before moving to another directory entry.

Fill

This button executes DADEs powerful Fill function. This allows you to fill regions of your disk with a specified character value or duplicate a copy buffer across a range of sectors. First you specify the starting and ending sector you want the fill. You can then further restrict the fill by specifying starting and ending offsets within each sector to fill. Then you specify what you want to fill the region with.



If you specify a byte value then each byte within the specified fill region will be replaced by the indicated byte. You can also choose to have the contents of the copy buffer copied into the specified region. This is an easy way to duplicate a large number of sectors.

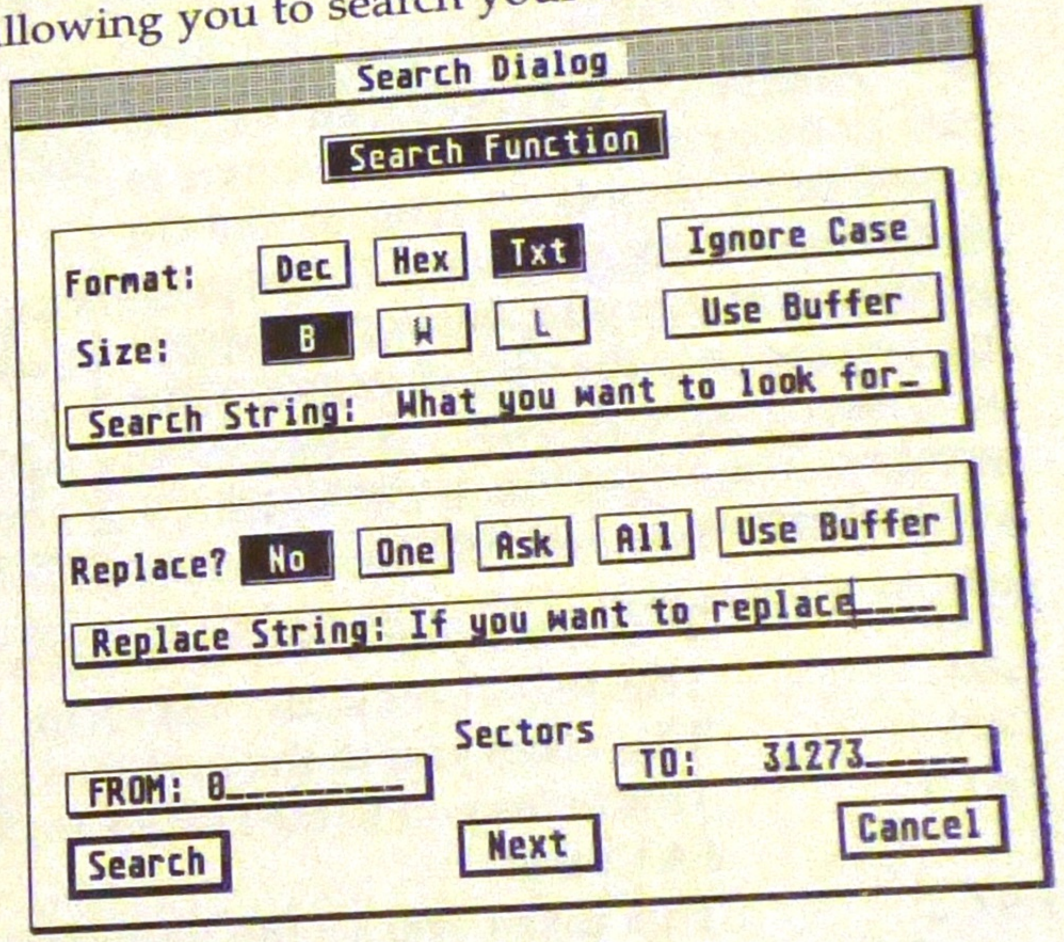
✔ Note that if you Double-click on the Fill button, it will clear the current edit buffer and fill the contents with zeros.

Print

This function prints the current edit buffer to your printer. The offset, data, and ASCII character columns are included on the printout.

Find

This function invokes DADEs powerful search function. The Find dialog will be displayed allowing you to search your disk for specified strings of data. You may specify the search be made on decimal numbers, hexadecimal numbers or ASCII text.



If searching by Decimal or Hex, you may restrict the search to byte, word, or long word aligned values.

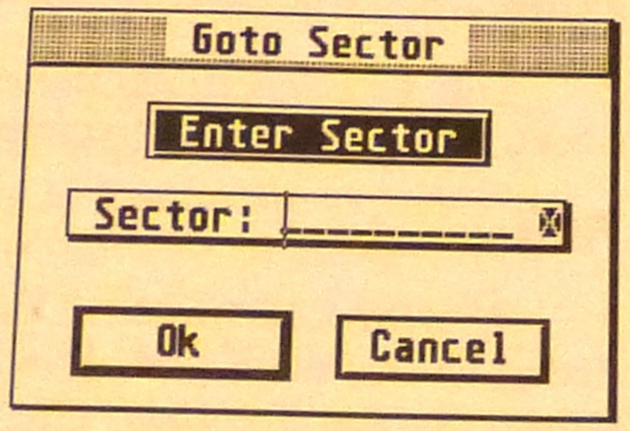
If searching for ASCII text, you may choose to ignore the case while searching. You may also specify that the contents of the copy buffer be used as the search string.

You may also specify whether to replace the found object(decimal, hex, or string value) with a specified replace value. You may choose not to replace, to replace one, all, or be asked each time the search value is found. You may also restrict the search to a specified starting and ending sector. This is useful to speed up searches when you know where you are looking, but are not quite sure of the search string to use.

To actually perform the specified search, click on the *Search* button. Complex searches on large drives can take a long time to perform. A status bar is displayed during the search to show you the progress. You may abort the search at any time by pressing the Escape key.

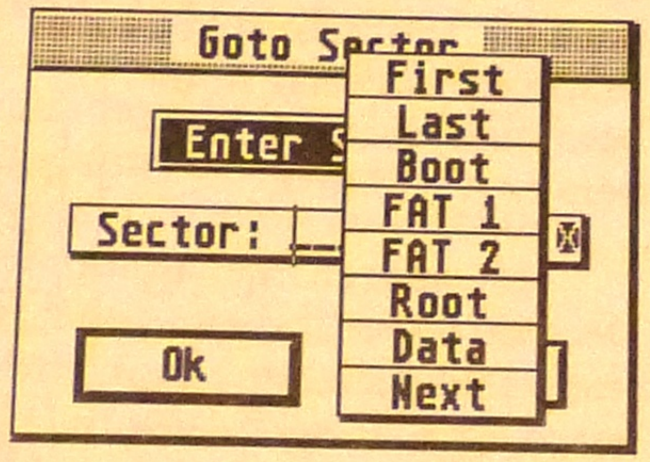
✔ To find the next occurrence of the search value, click on the *Next* button, and click on *Cancel* to exit the Search dialog. Note that double clicking on the Find button, performs the Search Next function with the last entered search criteria without having to open up the Find dialog.

Go To



The Go To function is a very powerful method of quickly moving around your disk. Clicking on the Go To button brings up the Go To dialog. You can then enter a specific sector to go to by entering the sector number in the editable text field.

✔ Alternatively, a pop-up list of short-cuts to frequently edited disk sectors is available by double clicking on the Go To sector editable text field. The pop-up list shown at the right will appear allowing you move to the disk structure without having to know the precise sector number. The following destinations are available from the Go To pop-up:



First	This takes you to the first sector on the disk. i.e. sector 0
Last	This takes you to the last sector on the disk.
Boot	This takes you to the partition Boot Sector
FAT 1	This takes you to the first sector of the first copy of the first FAT table.
FAT 2	This takes you to the first sector of the first copy of the second FAT table.
Root	This takes you to the first sector of the root directory.
Data	This takes you to the first data sector on the disk that actually contains data.
Next	This is a very useful function that searches your disk and takes you to the next sector that looks like it is a directory sector. Absolutely indispensable for manually recovering data from a disk whose Root directory or other important control structure information has been corrupted. You may execute this function without entering the Go To dialog by double clicking on the main Go To button.

To exit the pop-up without moving to one of the entries, click outside the pop-up box.

When a Go To is executed, the specified sector is loaded into the edit buffer and the current sector information is updated. If the current edit buffer has been modified, you will be prompted on whether you want to save your changes before going to the specified sector.

Append

This function allows you to append the current edit buffer to a file on the disk. A GEM file selector will appear to specify the file to append. If the file does not exist, type in a new name and the file will be created. This is an extremely useful function to manually recover lost files from a badly damaged disk. You can manually "walk" around your disk and hand rebuild lost files. Although this can be a tedious proceess, it is often the only way to recover data from a badly damaged disk. If the data is important enough to recover, you can recover it using this function.

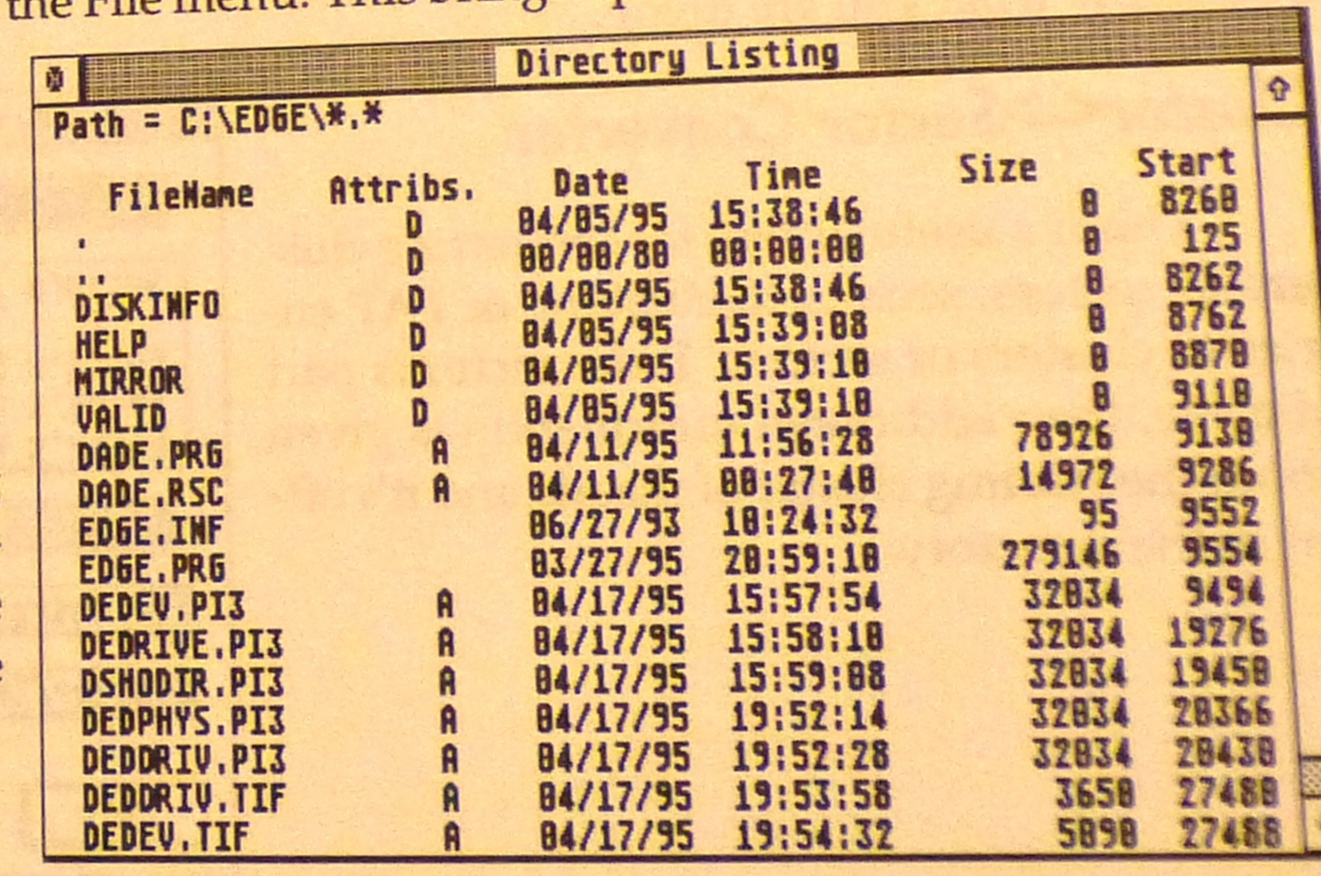
Write

✔ As was mentioned earlier, the edit buffer contains a copy of the current sector. No changes are made to the disk until you select Write. Selecting write will bring up a dialog asking you where to write the current edit buffer. This allows you to write the data to a different sector than it originated from and is useful if you are using the data in one sector as a template to write in other sectors. If you double click on the Write button, then the contents of the current edit buffer will be written to the current sector.

Other Functions

View Directory

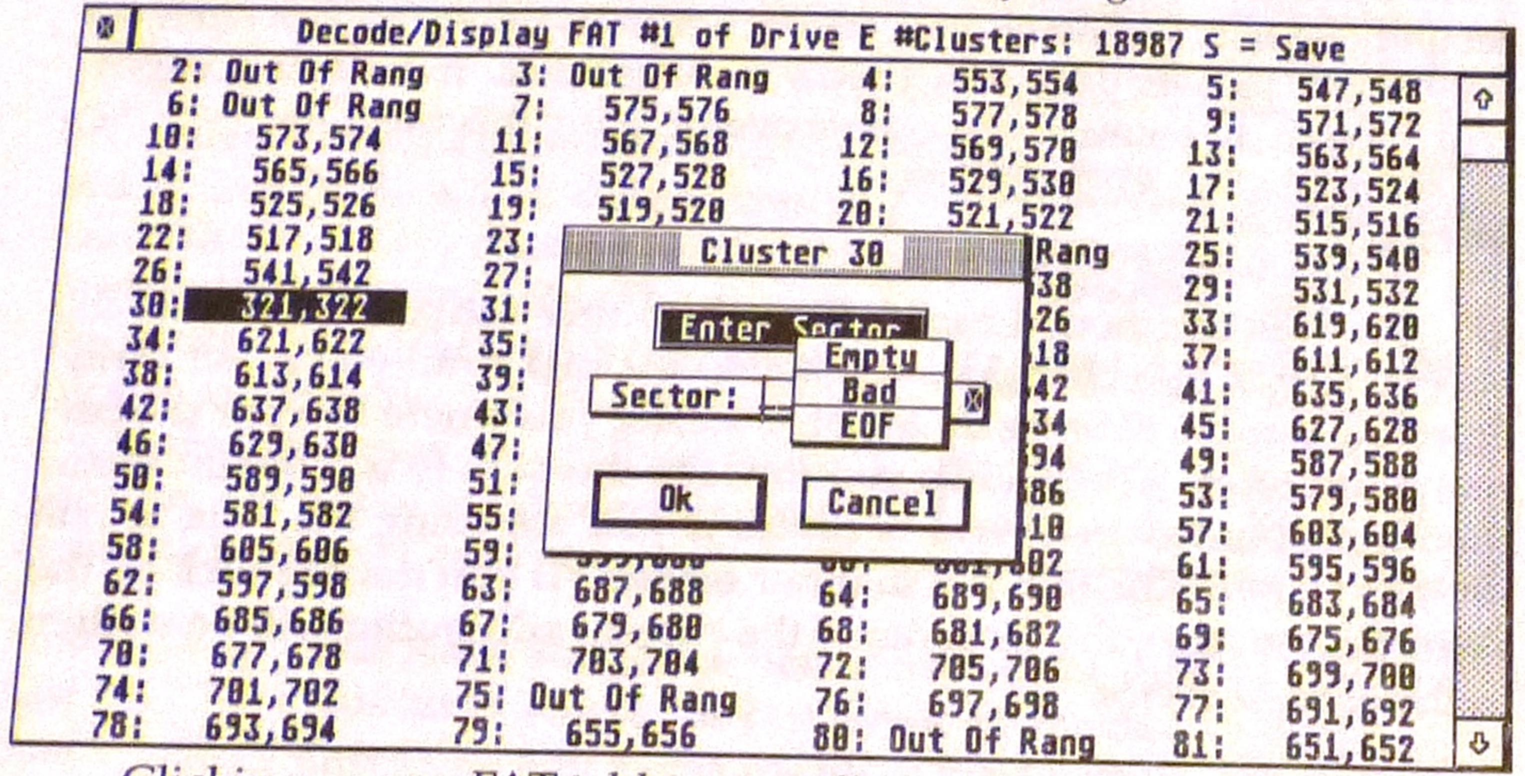
This function will display a decoded listing of any directory on your disk. This gives the file name, attributes, date, time, size and starting cluster of every entry in the directory. To execute this function, select *View Dir* from the File menu. This brings up a GEM file selector allowing you to specify the directory you want to view decoded. This is a very useful method of determining where specific files are located on the disk.



FileName	Attribs.	Date	Time	Size	Start
.	D	04/05/95	15:38:46	0	8260
..	D	00/00/00	00:00:00	0	125
DISKINFO	D	04/05/95	15:38:46	0	8262
HELP	D	04/05/95	15:39:08	0	8762
MIRROR	D	04/05/95	15:39:10	0	8870
VALID	D	04/05/95	15:39:10	0	9110
DADE.PRG	A	04/11/95	11:56:28	78926	9130
DADE.RSC	A	04/11/95	00:27:40	14972	9286
EDGE.INF		06/27/93	10:24:32	95	9552
EDGE.PRG		03/27/95	20:59:10	279146	9554
DEDEV.PI3	A	04/17/95	15:57:54	32034	9494
DEDRIVE.PI3	A	04/17/95	15:58:10	32034	19276
DSHODIR.PI3	A	04/17/95	15:59:08	32034	19458
DEDPHYS.PI3	A	04/17/95	19:52:14	32034	20366
DEDDRIV.PI3	A	04/17/95	19:52:28	32034	20438
DEDDRIV.TIF	A	04/17/95	19:53:58	3650	27488
DEDEV.TIF	A	04/17/95	19:54:32	5090	27488

Display/Edit FAT

This function displays a decoded window containing the FAT table for the currently selected logical drive. The contents of each FAT table entry are displayed along with the individual sector numbers contained in the cluster pointed to by the FAT table entry. An bad FAT entry with an illegal cluster number is indicated by *Out of Range.*

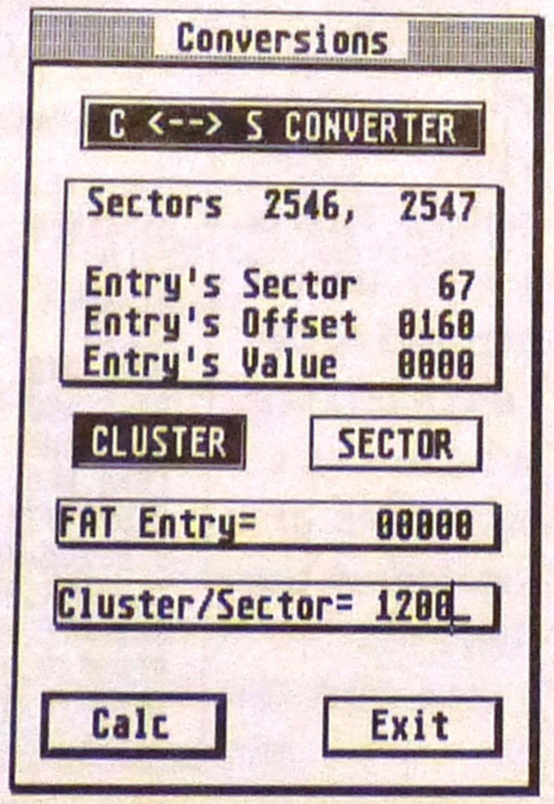


Clicking on any FAT table entry allows you to edit any individual FAT table entry. If you double click on the editable text field in the edit FAT dialog, a pop up list with commonly used entries for empty clusters, bad clusters, and End of File will be displayed. This saves time having to type in these commonly used values.

✔ *Warning!* It is important to not that this in an advanced function designed for experts only. Do not use this function unless you are sure you know what you are doing.

Cluster<->Sector Converter

This is a useful utility for converting clusters to sectors, sectors to clusters, or FAT entries to clusters or sectors. If the sector is part of a file, then additional information is given about the starting cluster of the file and it's offset in the directory.



Options

The following options can be set that control how DADE functions:

Edit R/O Files Selecting this option allows you to edit files whose attributes are set to read only.

Use FAT # This tells DADE to use either FAT #1 or FAT #2 when any program function that requires referencing the FAT table such as displaying file FAT chains etc.

Text size You may have DADE display it's data in either 9 or 10 point text sizes.

Centered This controls the location where dialog windows are placed. They can either be displayed in the center of the screen or at the position of the mouse pointer when the option was selected.

Cursor Type You may select the edit cursor to be constant or blinking.

Save Config This allows you to save the status of the options so they are set every time you run DADE.

Hotkeys

For ease of use, keyboard equivalents and hotkeys are provided for every major program function. In addition there are special keyboard commands that control the operation of various program functions.

During functions that scroll data such as Creating a Validation file or Validating files you may pause the display by selecting Control-S, pressing Control-Q will restart the process. Control-C is the universal abort key and applies to every abortable function, such as Mapping Bad Sectors or Validating a disk drive. Optimizing and the Disk Medic functions are not abortable via Control-C.

Basic Window Button Bar

Change Active Drive	Drive Letter
Disk Information	ALT-I
Fragmentation Map	ALT-M
Test Disk	ALT-T
Optimize Disk	ALT-O
Undelete Files	ALT-U
Run DADE	ALT-D
All Disk Information	ALT-A
Print	ALT-P

File Menu

Preferences	ALT-F
Open Window	ALT-W
Quit	Control-Q

Medic Menu

Test Disk Structure	ALT-T
Test Multiple	Control-T
Map Bad Sectors	ALT-B
Map Bad Multiple	Control-T
Create CRC File	ALT-R
Create Checksum File	ALT-C
Validate Files	ALT-V
Validate Multiple	Control-V
Partition Hard Disk	ALT-H

Optimize Menu

Optimize Drive	ALT-O
Optimize Multiple	Control-O

Undelete Menu

Configure MIRROR	ALT-N
Save Disk Info	ALT-S
Restore Disk Info	ALT-Y
Save SCSI Info	Control-S
Restore SCSI Info	Control-Y
Undelete Files	ALT-U

Utility Menu

Copy Drive Defrag	Control-C
Copy Drive Image	Control-I
Force Media Change	Control-M
View File	Control-F
Print File	Control-P
Browse Disk	Control-R
Zero Disk	Control-Z
Unzero Disk	Control-X
Wipe Partition	Control-W

Help Menu

File Help	F1
Medic Help	F2
Optimize Help	F3
Undelete Help	F4
Utility Help	F5
DADE Help	F6

Technical Support

Oregon Research's technical support staff is trained to provide you with fast and courteous service. Your purchase of Diamond Edge comes with 90 days of free technical support via letter, phone, FAX, and electronic mail. After 90 days free technical support is still available, but only via postal letter. Please return your completed owners registration card to become eligible for technical support.

If you require technical support via phone, FAX, or electronic mail beyond the 90 day free support period, extended technical support plans are available for a nominal charge. Contact Oregon Research in North America or HiSoft in Europe for more information on extended support plans.

In North America contact:

Oregon Research Associates
16200 S.W. Pacific Hwy., Suite 162
Tigard, OR 97224
USA

Phone: (503) 620-4919
FAX: (503) 624-2940

Electronic Support:

Genie: ORA
CompuServe: 71333,2655
Internet: orres@teleport.com

In Europe contact:

HiSoft
The Old School
Greenfield, Bedford
UNITED KINGDOM MK45 5DE

Phone: +44 525 718181
FAX: +44 525 713716

Electronic Support:

Internet: hisoft@cix.compulink.co.uk

Index

HiSoft SYSTEMS
The Old School, Greenfield
Bedford MK45 5DE UK
Tel: +44 (0) 1525 718181
Fax: +44 (0) 1525 713716

Hardware Warranty Registration

Complete and return this hardware warranty within 30 days of purchasing a HiSoft Systems' hardware product, postage is pre-paid if posted within the United Kingdom. Failure to do so may void the warranty. We will also use your name and address to send you mail periodically, informing you of our latest releases, software upgrades etc. If you do not wish to receive this mail, please tick the box below.

When entering the details for your HiSoft Systems' product please print your details clearly using block capitals wherever possible.

Product & Model/Version: ______________________

Serial Number: ______________________

Computer: ______________________

Purchased From: ______________________

____

Date Puchased: ______________________

Name: ______________________

Company: ______________________

Address: ______________________

____

____

Phone Number: ______________________

Warranty period is 12 months from date of purchase

☐ I do not wish to receive any further mail from HiSoft Systems

Detach here and Return

HiSoft Systems' Warranty Conditions

Software and User Guide warranty

HiSoft Systems will replace defective distribution media or user guides at no charge, provided you return the item to be replaced with proof of purchase to HiSoft Systems within the 30 day software and user guide warranty period. All implied warranties on the media and user guide, including implied warranties on merchantability and fitness for a particular purpose, are limited in duration to the software and user guide warranty period. The software and user guide warranty period begins with the date of the original purchase of this product.

Extended Software Support

All HiSoft Systems software products are covered by a 30-day free technical support period from receipt of the Software Warranty Registration card correctly filled out. Thereafter, if you wish to continue to receive technical support, there is a choice of two extended software support services; *Silver* and *Gold*.

Silver Support

Silver support offers you the following extended services:

- Lifetime technical help by letter, fax or telephone (during our technical support hour, 2-3 pm local UK time).
- Free periodic newsletters detailing new products and upgrades to existing titles.
- Access to HiSoft's closed user support conference on the CIX conferencing system.

Silver technical support covers all the HiSoft Systems' software products owned and registered by you for a single lifetime fee (per computer platform) of only £19.95 within Europe or only £29.95 outside Europe.

Gold Support

Gold support offers you the following extended services:

- Silver support.
- Free initial registration with the CIX electronic mail service.
- Automatic inter-version upgrade service. This means that you will be sent all intermediate upgrades for a product, free of charge; naturally, this does not cover major version changes such as Devpac version 2 to Devpac version 3.

Gold service is charged on an annual, per user basis, covering all HiSoft Systems software products owned and registered by the Gold service customer. The price is £39.95 inclusive per annum within Europe and £49.95 outside Europe.

Joining one of the extended software support schemes couldn't be easier, simply phone HiSoft free on 0500 223660 (UK only) and quote your credit/debit card details to one of our sales staff. Payments by cheque/postal order should be made payable to 'HiSoft'.

Detach here and Return

Software Warranty Registration

HiSoft SYSTEMS
The Old School, Greenfield
Bedford MK45 5DE UK
Tel: +44 (0) 1525 718181
Fax: +44 (0) 1525 713716

Please complete and return this software registration card within 30 days of purchasing a HiSoft Systems' software product; postage is pre-paid if posted within the United Kingdom. On of receipt of this card, you will be entitled to 30 days free technical support. We will also use your name and address to send you mail periodically, informing you of our latest release, software upgrades etc. If you do not wish to receive this mail, please tick the box below. We will not pass on your details to other companies.

When entering the details for your HiSoft Systems' product please print your details clearly using block capitals wherever possible.

Product & Model/Version: ______________________

Serial Number: ______________________

Computer: ______________________

Purchased From: ______________________

____

Date Puchased: ______________________

Name: ______________________

Company: ______________________

Address: ______________________

____

____

Phone Number: ______________________

☐ I do not wish to receive any further mail from HiSoft Systems

BUSINESS REPLY SERVICE
Licence No BF975

2

HiSoft Systems
The Old School
Greenfield
Bedford
MK45 5BR

HiSoft Systems' Warranty Conditions

Hardware warranty

On receipt of the Hardware Warranty Registration card correctly filled out, HiSoft Systems warrants to the original purchaser that its hardware products are free from defects in materials and workmanship for the period of 12 months after the purchase date. If you report such a defect, during the warranty period, HiSoft Systems will, at its option, repair or replace the product at no extra charge.

This warranty is void if the product has been damaged by accident, abuse, misuse or mis-application, has been serviced or modified without the written permission of HiSoft Systems or if any HiSoft Systems' serial number or warranty sticker has been removed or defaced. HiSoft Systems may use re-conditioned or re-furbished parts in making warranty repairs.

General warranty information

In no event will HiSoft Systems be liable for direct or indirect, special, incidental or consequential damages resulting from product failure or defect or breach of warranty under any legal theory, including lost profits, downtime, goodwill, damage to or replacement of equipment & property and any costs of recovering, reprogramming or reproducing any problem or data stored or used with HiSoft Systems' products.

The warranty set forth is exclusive of all others, oral, written, expressed or implied. No HiSoft Systems' dealer, agent or employee is authorised to extend, modify or add to this warranty.

Conditions

The warranty applies to the original purchaser and is not transferable.

The warranty will be invalidated immediately by any unauthorised alteration or repair to the product.

The liability of HiSoft Systems is limited to the cost of and/or replacement of the unit under warranty. HiSoft Systems do not accept any liabilities for loss of income, orders or time due to product failure.

Before implementing a return under these warranty terms, the user shall first ascertain that it is the unit which is defective. If HiSoft Systems determines the product has no hardware fault or has been returned outside the warranty condition laid out, HiSoft Systems will notify the user and may request return shipment at its own cost with HiSoft Systems' standard labour charge for services rendered.

This warranty does not cover fair wear and tear. In addition, accidental damage, misuse, incorrect line voltage, damage caused by improper installation, unauthorised repair, faulty attached equipment, abuse and altered serial numbers do not come within the terms of the warranty.

This warranty does not affect the purchaser's statutory rights and is in addition to any original manufacturer's warranty that may apply to component parts. This warranty is valid in the UK only.

Please telephone to obtain a *Returns Authorisation Number* (RAN) before returning the goods for repair, in their original packaging, postage paid.

BUSINESS REPLY SERVICE
Licence No BF975

2

HiSoft Systems
The Old School
Greenfield
Bedford
MK45 5BR

Sender:

Brandt Fernandes (Europe) Ltd.
The Desktop Centre
17–19 Blackwater Street
London SE22 8RS

Product Registration Card

Name: ____

Address: ____

____

________________________________ Postcode: ____________

Tel: ____________________ Fax: ____________________

Name of product purchased (one per card): ____________________

Serial number: ____________________ Date purchased: ____________

Where did you purchase this product? ____________________________

Where did you hear about it?

- ☐ CIX
- ☐ BBS
- ☐ Genie
- ☐ magazine ad
- ☐ review
- which magazine? ____________________
- ☐ computer store
- ☐ other ____________________________________

Comments: ____