CIS-24 Home http://www.c-jump.com/CIS24/CIS24syllabus.htm

The FAT File System

---

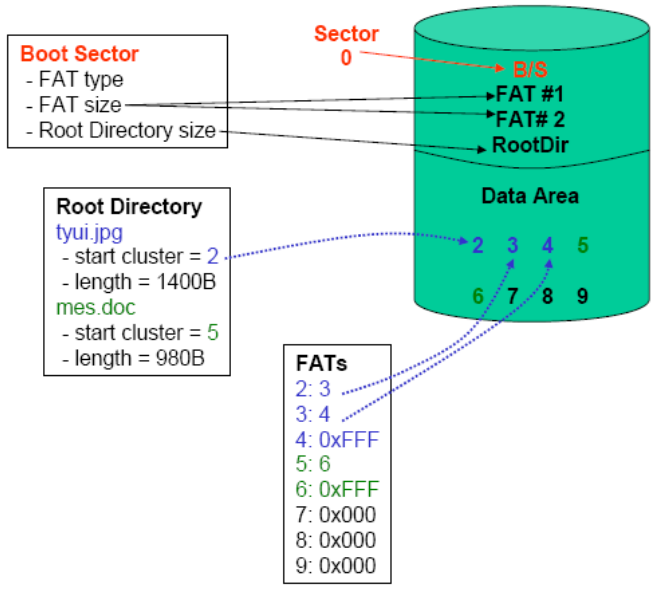
1. FAT Overview

- Simple - and common - file system
- Found on all Windows OS and many devices
  - **FAT12**: Developed 1977 (MS Disk BASIC)
  - **FAT16**: Developed 1987 (MS-DOS 3.31)
  - **FAT32**: Developed 1996 (Win95 OSR2)
- Few data structures supported:
  - Cluster: Basic storage unit for files
  - Directory: Lists file name, starting cluster, and length
  - File Allocation Table: Contains cluster status and pointer to next cluster in chain

---

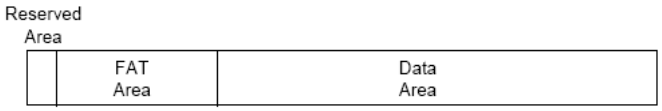
2. Boot Sector, FAT, Root Directory, and Files

- File `tyui.jpg`:
  - occupies clusters 2, 3, and 4.
  - The file size is 1,400 bytes, it occupies 1,536 bytes (3 clusters) on the disk, and cluster 4 includes 136 bytes of slack space.
- File `mes.doc`:
  - occupies clusters 5 and 6.
  - The file size is 980 bytes, it occupies 1,024 bytes (2 clusters), and has 44 bytes of slack space in cluster 6.
- Clusters 7, 8, and 9 are unallocated.



---

3. FAT File System Layout



---

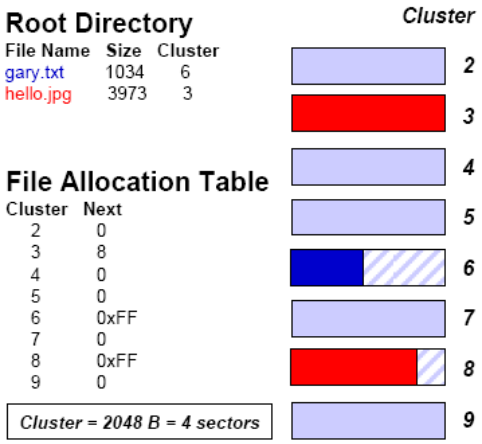
4. FAT Clusters and Sectors

- A cluster is a group of consecutive sectors
  - A sector is usually 512 B
  - A cluster is 1, 2, 4, 8, 16, 32, or 64 sectors (i.e., it can range from 512 B to 32 KB)
- Each cluster has an address
- The first cluster has an address of 2
  - I.e., there is no addressable cluster 0 or 1

---

5. FAT, Slack, and Unallocated Space

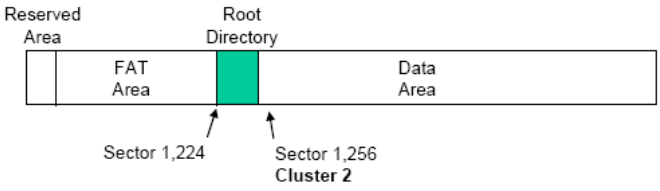
- Clusters 3, 6, and 8 are allocated; clusters 2, 4, 5, 7, and 9 are unallocated
- Clusters 6 and 8 are only partially filled; the unused portion is slack space
- File `gary.txt`:
  - logical size is 1,034 bytes
  - physical size is 2,048 bytes (slack = 1,014 B)
- File `hello.jpg`:
  - logical size is 3,973 bytes
  - physical size is 4,096 bytes (slack = 123 B)



6. Where is the First FAT Cluster?

- The first cluster is Cluster 2
- Actual location of cluster 2 is different in FAT12/16 and FAT32
- Assume cluster size = 2,048 B (4 sectors)
- Assume that data area starts at sector 1224
- First sectors of data area are reserved for the Root Directory
  - Size is established at boot time
- Cluster 2 starts after Root Directory
- Root directory is set at 32 sectors
  - Occupies sectors 1,224-1,255

FAT12/16 Cluster Example:



- Cluster 2 starts at sector 1,256
- Cluster 3 starts at sector 1,260
- Cluster 4 at 1,264...

7. Boot Sector

- First sector of a FAT system is the boot sector
  - Contains most of the information with which to determine
    - the file system type, and
    - size and location of data structures
- Boot sector format is different for FAT12/16 and FAT32

8. FAT Boot Sector, bytes 0-35 (FAT12/16 and FAT32)

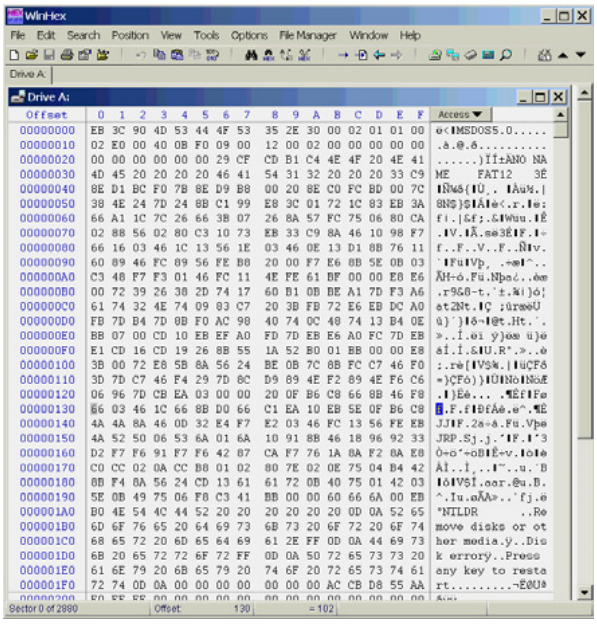
Bytes	Purpose
0-2	Assembly code instructions to jump to boot code (mandatory in bootable partition)
3-10	OEM name in ASCII
11-12	Bytes per sector (512, 1024, 2048, or 4096)
13	Sectors per cluster (Must be a power of 2 and cluster size must be <=32 KB)
14-15	Size of reserved area, in sectors

16	Number of FATs (usually 2)
17-18	Maximum number of files in the root directory (FAT12/16; 0 for FAT32)
19-20	Number of sectors in the file system; if 2 B is not large enough, set to 0 and use 4 B value in bytes 32-35 below
21	Media type (0xf0=removable disk, 0xf8=fixed disk)
22-23	Size of each FAT, in sectors, for FAT12/16; 0 for FAT32
24-25	Sectors per track in storage device
26-27	Number of heads in storage device
28-31	Number of sectors before the start partition
32-35	Number of sectors in the file system; this field will be 0 if the 2B field above (bytes 19-20) is non-zero

9. FAT Boot Sector (FAT12/16)

Bytes	Purpose
0-35	(See previous table)
36	BIOS INT 13h (low level disk services) drive number
37	Not used
38	Extended boot signature to validate next three fields (0x29)
39-42	Volume serial number
43-53	Volume label, in ASCII
54-61	File system type level, in ASCII. (Generally "FAT", "FAT12", or "FAT16")
62-509	Not used
510-511	Signature value (0xaa55)

10. FAT12 Boot Sector



11. Boot Sector Interpretation

```
00-02: eb 3c 90    Instructions to jump to boot code
03-0a: 4d 53 44 4f 53 35 2e 30
                   Name string (MSDOS5.0)
0b-0c: 00 02       Bytes/sector (0x0200 = 512)
0d   : 01          Sectors/cluster (1)
0e-0f: 01 00       Size of reserved area (1 sector)
10   : 02          Number of FATs (2)
11-12: e0 00       Max. number of root directory entries (0x00e0 = 224)
13-14: 40 0b       Total number of sectors (0x0b40 = 2,880)
15   : f0          Media type (removable)
16-17: 09 00       FAT size (0x0009 = 9 sectors)
18-19: 12 00       Sectors/track (0x0012 = 18)
1a-1b: 02 00       Number of heads (0x0002 = 2)
1c-1f: 00 00 00 00 Number of sector before partition (0)
20-23: 00 00 00 00 Total number of sectors (0 because 2B value not equal 0)
24   : 00          Drive number (0)
25   : 00          Unused
26   : 29          Extended boot signature
27-2a: cf cd b1 c4 Volume serial number (C4B1-CDCF)
2b-35: 4e 4f 20 4e 41 4d 45 20 20 20 20
                   Volume label ("NO NAME ")
36-3d: 46 41 54 31 32 20 20 20
                   File system type label ("FAT12 ")
3e-1fd : [snip]    Not used
1fe-1ff: 55 aa     Signature value (0xaa55)
```

---

12. Capacity of this Medium

- FAT12 allocates 12 bits per FAT entry
  - Limits addressing to 4,096 (212) clusters
- This (removable) device is configured so that:
  - 1 cluster = 1 sector
  - 1 sector = 512 B
- This FAT12 table is limited in capacity to 2,097,152 bytes (2 MB)
  - I.e., 4K clusters of 512 B each
- This device has 2,880 sectors
  - 512 B clusters yields a device capacity of 1.44 MB
  - Corresponds to what we would expect for a floppy

---

13. Sector Assignments

Sector(s)	Address	Function
0	0x0000-0x01ff	Boot Sector
1-9	0x0200-0x13ff	File Allocation Table (primary)
10-18	0x1400-0x25ff	File Allocation Table (secondary)
19-32	0x2600-0x41ff	Root Directory
33-2879	0x4200-0x167fff	File storage space

NOTES:

- Boot Sector is 1 sector (0x200 bytes)
- There are two FATs, each 9 sectors (0x1200 bytes)
- The Root Directory can contain 224 entries, each 32 bytes (7168, or 0x1c00, bytes; 14 sectors)
- File storage starts at sector #33 (1+9+9+14), byte #0x4200 (0x200+0x1200+0x1200+0x1c00)

---

14. Root Directory

- Contains file names and metadata
  - Located immediately after FAT(s) in FAT12/16 or in a location specified in the FAT32 boot sector
- Supports 8.3 names or long file names
- New entries are added to the directory using a first-available or next-available strategy
  - First-available: Finds first unallocated entry in the directory (e.g., Win98)
  - Next-available: Finds next available entry from the last allocated entry; at end of directory chain, start again at beginning (e.g., WinXP)

---

15. Root Directory Entries

- The Root Directory is a series of entries describing files
- Each entry is 32 bytes and contains
  - single short (8.3) filename (SFN),
  - attributes,
  - MAC times,
  - start cluster,
  - size,
  - and other metadata.
  - Additional 32B entries contain the file's long filename (LFN)

---

16. Root Directory Entry Format (SFN)

Root Directory SFN Entry Data Structure

Bytes	Purpose
0	First character of file name (ASCII) *or* allocation status (0x00=unallocated, 0xe5=deleted)
1-10	Characters 2-11 of the file name (ASCII); the "." is implied between bytes 7 and 8
11	File attributes (see File Attributes table)
12	Reserved
13	File creation time (in tenths of seconds)*
14-15	Creation time (hours, minutes, seconds)*
16-17	Creation date*
18-19	Access date*
20-21	High-order 2 bytes of address of first cluster (0 for FAT12/16)*
22-23	Modified time (hours, minutes, seconds)
24-25	Modified date
26-27	Low-order 2 bytes of address of first cluster
28-31	File size (0 for directories)

File Attributes

Flag Value	Description
0000 0001 (0x01)	Read-only
0000 0010 (0x02)	Hidden file
0000 0100 (0x04)	System file
0000 1000 (0x08)	Volume label
0000 1111 (0x0f)	Long file name
0001 0000 (0x10)	Directory
0010 0000 (0x20)	Archive

** Bytes 13-22 are unused by DOS*

---

17. Root Directory Example

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000FF120	42	70	00	67	00	00	00	FF	FF	FF	FF	0F	00	D3	FF	FF
000FF130	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF
000FF140	01	62	00	65	00	6C	00	69	00	6E	00	0F	00	D3	5F	00
000FF150	67	00	61	00	79	00	6C	00	65	00	00	00	2E	00	6A	00
000FF160	42	45	4C	49	4E	5F	7E	31	4A	50	47	20	00	96	CF	82
000FF170	FC	34	FC	34	00	00	31	B0	B6	32	8D	00	B6	6A	05	00
000FF180	42	6A	00	70	00	67	00	00	00	FF	FF	0F	00	56	FF	FF
000FF190	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF
000FF1A0	01	6B	00	65	00	73	00	73	00	6C	00	0F	00	56	65	00
000FF1B0	72	00	5F	00	67	00	61	00	72	00	00	00	79	00	2E	00
000FF1C0	4B	45	53	53	4C	45	7E	31	4A	50	47	20	00	AB	CF	82
000FF1D0	FC	34	FC	34	00	00	31	B0	B6	32	E4	00	29	6A	05	00
000FF1E0	E5	54	00	68	00	75	00	6D	00	62	00	0F	00	A4	73	00
000FF1F0	2E	00	64	00	62	00	00	00	FF	FF	00	00	FF	FF	FF	FF
000FF200	E5	48	55	4D	42	53	20	20	44	42	20	26	00	C1	CF	82
000FF210	FC	34	FC	34	00	00	D3	8D	DC	34	3B	01	00	20	00	00

Three files shown here:

```
BELIN_~1.JPG @ offset 0xff160 (belin_gayle.jpg entry starts @ offset 0xff140)
KESSLE~1.JPG @ offset 0xff1c0 (kessler_gary.jpg entry starts @ offset 0xff1a0)
?HUMBS.DB @ offset 0xff200 (Thumbs.db; deleted)
```

---

18. Sample Root Directory Entry

FAT Directory Entry, Base Offset: FF200

Record #: 144 Close

Offset	Title	Value	
FF200	Filename (blank-padded)	åHUMBS	
FF208	Extension (blank-padded)	DB	
FF20B	0F = LFN entry	26	
FF20B	Attributes [- -a-dir-vol-s-h-	00100110	
FF200	00 = Never used, E5 = Erased	E5	
FF20C	(reserved)	0	
FF20E	Creation date & time	07/28/2006	16:22:30
FF20D	Cr. time refinement in 10-ms u	193	
FF210	Access date (no time!)	07/28/2006	06:39:56
FF216	Update date & time	06/28/2006	17:46:38
FF214	(FAT 32) High word of cluster	0	
FF21A	16-bit cluster #	315	
FF21A	32-bit cluster #	315	
FF21C	File size (zero for a director	8192	

---

19. Another Sample Root Directory Entry

FAT Directory Entry, Base Offset: FF1C0

Record #: 142 Close

Offset	Title	Value	
FF1C0	Filename (blank-padded)	KESSLE~1	
FF1C8	Extension (blank-padded)	JPG	
FF1CB	0F = LFN entry	20	
FF1CB	Attributes [- -a-dir-vol-s-h-	00100000	
FF1C0	00 = Never used, E5 = Erased	4B	
FF1CC	(reserved)	0	
FF1CE	Creation date & time	07/28/2006	16:22:30
FF1CD	Cr. time refinement in 10-ms u	171	
FF1D0	Access date (no time!)	07/28/2006	06:39:56
FF1D6	Update date & time	05/22/2005	22:01:34
FF1D4	(FAT 32) High word of cluster	0	
FF1DA	16-bit cluster #	228	
FF1DA	32-bit cluster #	228	
FF1DC	File size (zero for a director	354857	

20. FATs Compared

Attribute	FAT12	FAT16	FAT32
Used For	Floppies; small hard drives	Small to large hard drives	Large to very large hard drives
Size of Each FAT Entry	12 bits	16 bits	28 bits
Maximum Number of Clusters	~4,096	~65,536	~268,435,456
Supported Cluster Sizes	512 B to 4 KB	2 KB to 32 KB	4 KB to 32 KB
Maximum Volume Size	16,736,256 B (16 MB)	2,147,123,200 B (2 GB)	~2^{41} B (2 TB)

21. FAT12 File Allocation Table

- FAT table entries are *packed* so that two cluster entries occupy three bytes with the following general format:

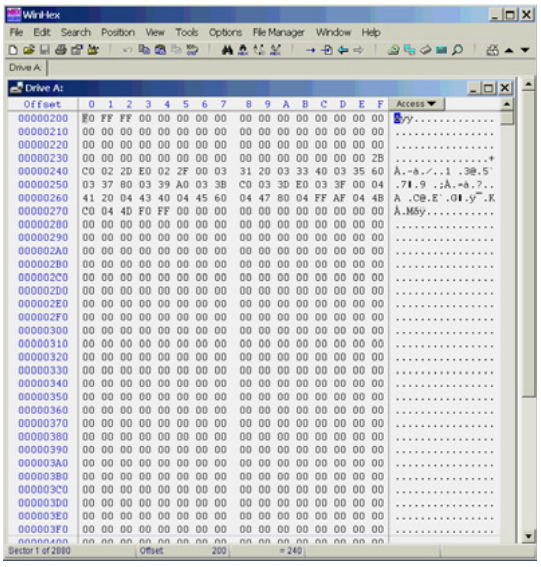
```
yz Zx XY
```

  where
  - xyz is the one pointer entry and
  - XYZ is the second pointer entry.
- E.g., bytes 242-244:

```
2d e0 02
```

  refer to clusters 0x02d (45) and 0x02e (46)
- Primary FAT starts at sector 1, byte 0x200 (shown here)
- The starting cluster in the directory is also a pointer into the FATs linking to the next cluster in the file

Primary FAT sector 1:



22. Interpreting FAT12

```
0260  41 20 04 43 40 04 45 60 04 47 80 04 ff af 04 4b  A .C@.E`.G..ÿ¯.K
0270  c0 04 4d f0 ff 00 00 00 00 00 00 00 00 00 00 00  À.Mðÿ............
```

- The Primary FAT starts at Sector 1, byte #0x0200.
- Suppose a file has a starting cluster of 0x49 (73).
  - The file starts on the media at cluster #73
  - The FAT entry pointing to the next cluster is in the FAT at byte #(73*1.5 + 1) = 110 (0x6e)

- Since the FAT starts at 0x0200, the FAT entry for this file is at 0x026e

- Example:

  1st cluster is 0x49 (73). FAT entry starts at high-order nibble of 0x026e (110) = 0x04a (74)

  2nd cluster is 0x4a (74). FAT entry starts at low-order nibble of 0x0270 (112) = 0x04b (75)

  3rd cluster is 0x04b (75). FAT entry starts at high-order nibble of 0x0271 (113) = 0x04c (76)

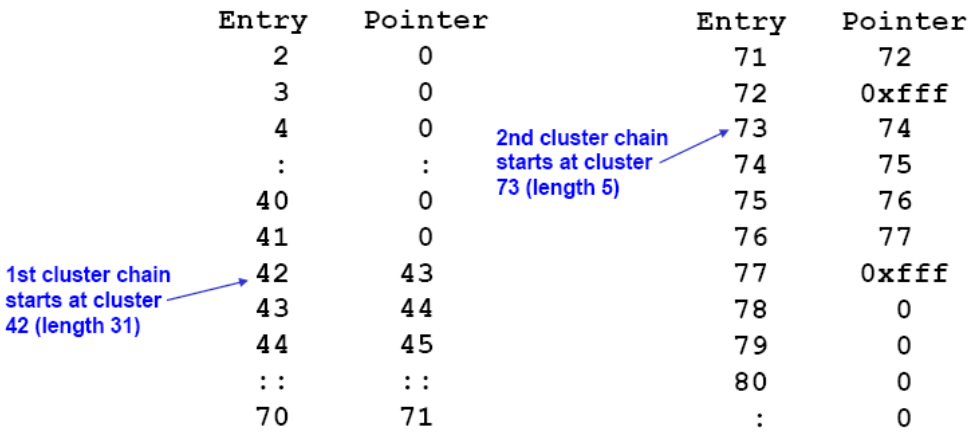
  4th cluster is 0x04c (76). FAT entry starts at low-order nibble of 0x0273 (115) = 0x04d (77)

  5th cluster is 0x04d (77). FAT entry starts at high-order nibble of 0x0274 (116) = 0xfff (end of list)

- The physical size of this file is five clusters (2560 bytes), and occupies clusters 73, 74, 75, 76, and 77 on the medium. (It is merely a coincidence that the clusters are contiguous.)

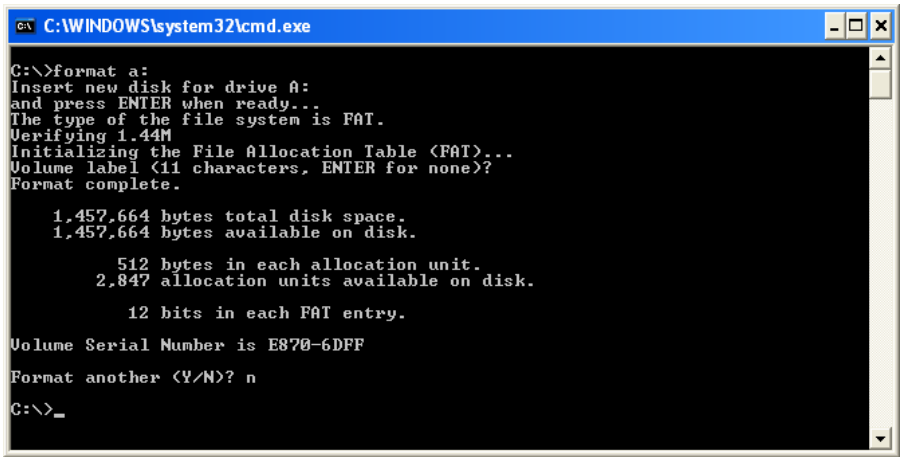
---

23. FAT12 Contents



---

24. Formatting a Floppy

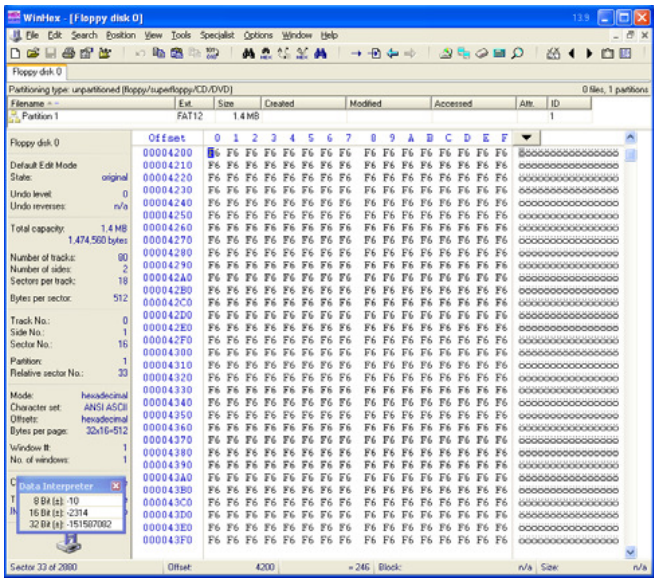


- Formatting will initialize the FATs (0x00) and root directory (0xF6)
- On a floppy, format will overwrite the data area with 0xF6
- Data is NOT deleted when using Quick Format on a floppy or any format on a hard drive

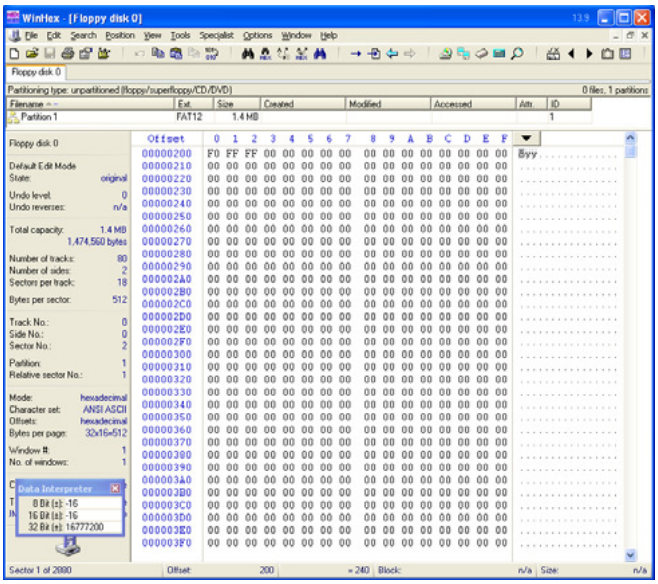
---

25. Formatted Floppy Data Structures

Floppy Data Structures before formatting, uninitialized:



Floppy Data Structures after formatting, initialized root directory:



26. Allocating A New File

1. Find first free entry in directory and write file name
2. Search FAT for unallocated cluster; set to EOF (0xFFF)
3. Write that cluster's address into directory entry
4. If another cluster is needed,
    - find an unallocated FAT entry,
    - reset that value to EOF, and
    - reset previous FAT pointer to this new cluster

   Repeat this step as necessary

27. Deleting A File

1. Find directory entry for file to delete
2. Using starting cluster value in the root directory, set all FAT entries in file's cluster chain to zero
3. Deallocate directory entry by overwriting first byte of the entry with 0xE5 (å)

28. For More Information...

- FAT: General Overview of On-Disk Format, v1.03 (12/6/2000), Microsoft:

  http://www.microsoft.com/whdc/system/platform/firmware/fatgen.mspx http://digitalforensics.champlain.edu/download/FAT_general_overview-LFN.pdf

- File Allocation Table, Wikipedia:

  http://en.wikipedia.org/wiki/File_Allocation_Table