

CS323 Operating Systems File System V

Yuanyuan Zhou
Lecture 28
4/7/2003

Content of this lecture

- Administrative announcements
- File system case studies
- Summary

2

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Review

- File system reliability
 - Data availability, integrity, consistency
 - Full vs. incremental backups
- Log structured file systems
 - Idea
 - Tradeoff with traditional file systems

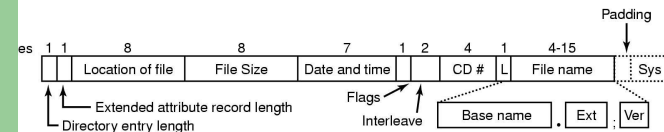
3

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Example File Systems CD-ROM File Systems

- The ISO 9660 directory entry

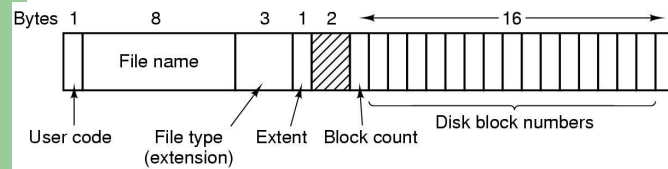


4

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The CP/M File System: ancestor of MS-DOS



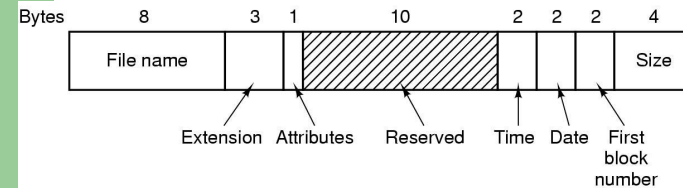
The CP/M directory entry format

5

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The MS-DOS File System (FAT-16)



The MS-DOS directory entry
Why is it called FAT-16?
Disadvantage?

6

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The MS-DOS File System (2)

Block size	FAT-12	FAT-16	FAT-32
0.5 KB	2 MB		
1 KB	4 MB		
2 KB	8 MB	128 MB	
4 KB	16 MB	256 MB	1 TB
8 KB		512 MB	2 TB
16 KB		1024 MB	2 TB
32 KB		2048 MB	2 TB

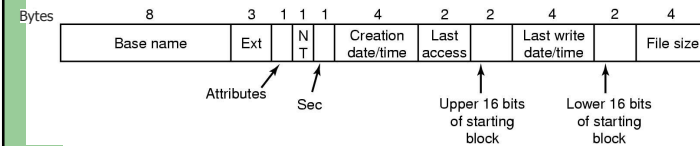
- Maximum partition for different block sizes
- The empty boxes represent forbidden combinations

7

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The Windows 98 File System (1)



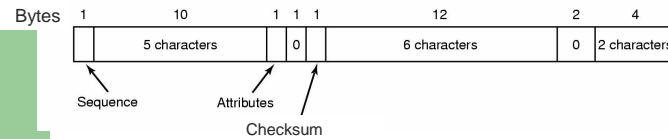
The extended MOS-DOS directory entry used in Windows 98

8

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The Windows 98 File System (2)



An entry for (part of) a long file name in Windows 98

9

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The Windows 98 File System (3)

68	d	o	g		A	0	C	K					0		
3	o	v	e		A	0	C	K	t	h	e		l	a	0
2	w	n		f	o	A	0	C	x		j	u	m	p	0
1	T	h	e		q	A	0	C	u	i	c	k		b	0
T	H	E	Q	U	I	~	1	A	N	T	S	Creation time	Last acc	Upp	Last write
												Low	Size		

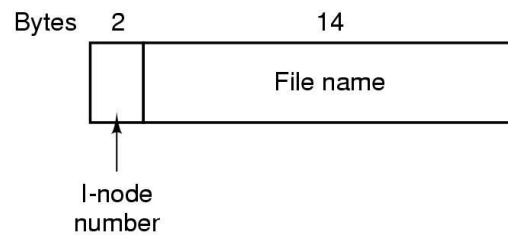
An example of how a long name is stored in Windows 98

10

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The UNIX V7 File System (1)



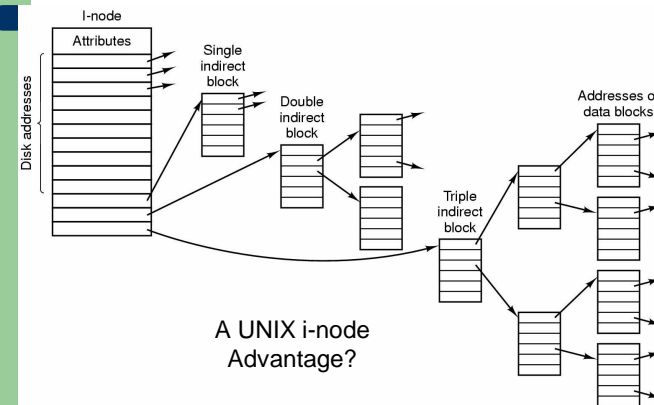
A UNIX V7 directory entry

11

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The UNIX V7 File System (2)



12

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

The UNIX V7 File System (3)

Root directory	i-node 6 is for /usr	Block 132 is /usr directory	i-node 26 is for /usr/ast	Block 406 is /usr/ast directory
1 .	Mode	6 .	Mode	26 .
1 ..	size	1 ..	size	6 ..
4 bin	times	19 dick		64 grants
7 dev	132	30 erik	406	92 books
14 lib		51 jim		60 mbox
9 etc		26 ast		81 minix
6 usr		45 bal		17 src
8 tmp				
Looking up usr yields i-node 6	i-node 6 says that /usr is in block 132	/usr/ast is i-node 26	i-node 26 says that /usr/ast is in block 406	/usr/ast/mbox is i-node 60

The steps in looking up `/usr/ast/mbox`

CS 323 - Operating Systems,
Yuanyuan Zhou

4/6/2003

13

Group discussion

- If ONLY the root inode is in memory, what is the minimum number of disk accesses to read the first block of
– `/home/bob/mbox` ?

CS 323 - Operating Systems,
Yuanyuan Zhou

4/6/2003

14

Levels of Access Methods

- device level read and write sectors or tracks from disk
- the i/o is written at the level of a sequence of transfer commands to the controller
- this is often performed as if the access path is a channel

CS 323 - Operating Systems,
Yuanyuan Zhou

4/6/2003

15

Levels of Access Methods

- block level access to a file is in terms of blocks or physical records within a file
- the user must do his own buffering. Access methods include:
 - `Read(file, block_no)`
 - `Write(file, block_no)`
 - `Wait(file, block_no)`

CS 323 - Operating Systems,
Yuanyuan Zhou

4/6/2003

16

Levels of Access Methods Continued

- file level access to the file is in terms of acquiring access to a copy of the file that is stored in primary memory
- queued or buffered level access to the file is in terms of logical records that depend on software interpretation. for example, read and write chars in UNIX. buffering is used to provide logical record abstraction and maps i/o into physical records

17

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Levels of Access Methods Continued

- memory mapped file level
 - the file is mapped into virtual memory
 - file access is at the instruction level
 - page faults may read a page of file data from disk to memory
 - an address of a logical record within a file is given by a virtual memory address offset of that record from the beginning of the file

18

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Levels of Access Methods Continued

- persistent object
 - the file is mapped into virtual memory and
 - access to the contents of the file is provided by an abstract data type interface that is determined by
 - the type of the data stored in the file

19

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Protection

- in file systems, protection is needed from physical damage (reliability) and improper access (protection)
- reliability (chapter 12) is generally provided by duplicate copies of files

20

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Protection

- protection - various mechanisms for single-user system and multi-user systems
 - removing the floppy disk,
 - prohibiting access to files of other users

21

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Types - Controlled Access

- read - possible access to read from file
- write - possible access to write to a file
- execute - load file and execute it
- append - write new information at the end of a file
- delete - delete file and free its space for possible reuse
- list - list name and attributes of a file

22

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Access Lists and Groups

- associate each file and directory with access list
- problem with access list: length
- solution: condensed version of the access list
 - owner - user who created the file
 - group - a set of users who are sharing the file and need similar access
 - universe - all other users

23

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Access Lists Example

- UNIX - 3 fields of length 3 bits are used. fields are
- user(u),group(g),others(o), bits are read(r), write(w), execute(x) -
- example
- % Chmod go+rw file

24

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

UNIX Protection Scheme Detail

- *setuid* if file is executed, execution assumes the user id
- *setgid* if file is executed, it execution the group id
- *stickybit* while in use, text of program is not to be paged out
- *rw* read, write, execute owner permission.
- *rw* read, write, execute group permission.
- *rw* read, write, execute other permission.
- Files can be hidden from "group and others" by creating a directory that is executed but not readable "chmod 711 hidefiles" and creating the file in that directory. Users in the group or in others may access the file if they know the filename.

25

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Other Protection Approaches

- associate a password with each file
- protect directories - listing of file names might be a protected operation
- encryption

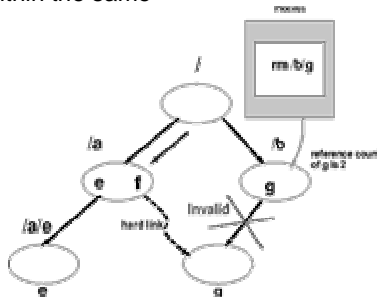
26

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Hard links

- Hard links are reference counted and can only be within the same file system.



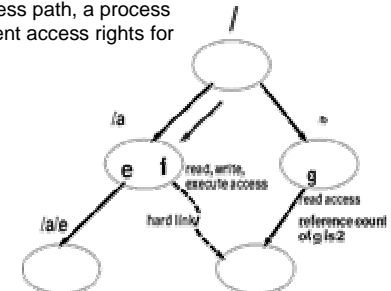
27

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Access Rights and Links

- Permissions along a path may prevent access but hard links can circumvent permissions.
- Depending upon the access path, a process may be able to get different access rights for a file.



28

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Symbolic Links

- Allow sharing of files and support any graph directory structure topologies.
- Access rights through a symbolic link do not depend upon the user access path:
 - the path that counts is the one that's stored in the symbolic link (`ln -s xyz link -- ls -l`).
- A file at the end of a symbolic link can be removed and leaves a dangling pointer.
- Cyclic file directories can be created
 - `tar` and other tools can be confused by symbolic links -- use the right options.

29

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Consistency Semantics

- UNIX
- Session Semantics
- Immutable-Shared File Semantics

30

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

UNIX Files

- writes to file immediately visible to other users reading from file
- can share current location into file through open file tables

31

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Session Semantics

- writes to open file by user not visible immediately – only to new opens.
- closing a write file doesn't make contents visible if applications have file open already

32

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Immutable-shared Files

- files, once shared, never change
- name may not be reused
- good in distributed system

33

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou

Summary

- file concepts - file attributes, operations, structures
- directory systems - single-level, two-level, tree structure, acyclic structure, general structure
- access methods - sequential, direct, indexed
- protection - possible access protection, access lists

34

4/6/2003

CS 323 - Operating Systems,
Yuanyuan Zhou