

OPERACIJSKI SUSTAVI
(predavanja u ak. god. 2006/07.)

doc.dr.sc. Goran Martinović
www.etfos.hr/~martin
goran.martinovic@etfos.hr
Tel: 031 224-766
Soba: 0-14

Osijek, 2007.

Datotečni sustav

- 6.1 Datoteke
- 6.2 Direktoriji
- 6.3 Implementiranje datotečnog sustava
- 6.4 Primjeri

Dugotrajno spremanje informacija

Tri osnovna zahtjeva za dugotrajno spremanje informacija

- Mora biti moguće spremiti i vrlo velike količine informacija
- Informacije moraju preživjeti završetak procesa koji ih je koristio
- Više procesa mora moći istovremeno pristupati informacijama

Rješenje: pohrana informacija na disk ili neki drugi vanjski medij u jedinice zvane **datoteke**.

Procesi iz njih mogu čitati, te pisati u njih. Informacije pohranjene u datoteke moraju biti postojane i nezavisne o stvaranju i završetku procesa. Datoteka može nestati samo kada to želi njen vlasnik.

O datotekama brine operacijski sustav (datotečni sustav):

- strukturiranje
- imenovanje
- pristup
- uporaba
- zaštita
- implementiranje

6.1. Datoteke

6.1.1. Dodjeljivanje imena datoteka

Datoteke - apstrakcijski mehanizmi (npr. sakriti od korisnika način spremanja informacija i način rada HDD)

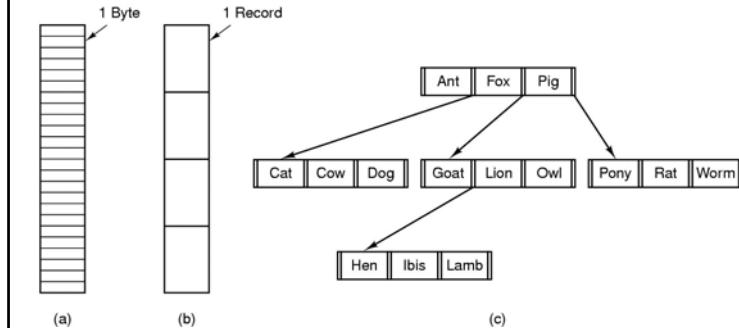
- rukovanje i imenovanje objekata (kada proces kreira datoteku, dodijeli joj ime, a kad proces završi, datoteka ostaje dostupna drugim procesima)
- imena: 8 znakova, 255 znakova; razlika velika i mala slova (UNIX)
- Windows 95, 98 – MS-DOS datotečni sustav
- Windows NT, 2000, XP – MS-DOS i NTFS (imena u Unicode)
- ime.proširenje (extension): ime.c; prog.c.Z
- imena mogu biti dogovori, ali neki programi na njima mogu inzistirati (compile)
- značenje ekstenzije moguće je dodavati u OS - Windows

Najčešći tipovi datoteka

Extension	Meaning
file.bak	Backup file
file.c	C source program
file.gif	Compuserve Graphical Interchange Format image
file.hlp	Help file
file.html	World Wide Web HyperText Markup Language document
file.jpg	Still picture encoded with the JPEG standard
file.mp3	Music encoded in MPEG layer 3 audio format
file.mpg	Movie encoded with the MPEG standard
file.o	Object file (compiler output, not yet linked)
file.pdf	Portable Document Format file
file.ps	PostScript file
file.tex	Input for the TEX formatting program
file.txt	General text file
file.zip	Compressed archive

5

6.1.2. Struktura datoteke



Tri vrste datoteka

- slijed bajta (bilo što unutra, fleksibilno, OS ne ograničava, ali ne može ni pomoći kod nedozvoljenih radnji) – (a)
- slijed slogova (jednake duljine, svaki sa svojom unutarnjom strukturom, čitanje i pisanje na razini sloga) – (b)
- stablo (datoteka je stablo slogova, moguće nejednakih duljina sa ključem na fiksnom mjestu u slogu, brzo pretraživanje po ključu, stvaran položaj datoteke nebitan, OS određuje smještaj, a ne korisnik) – (c)

6

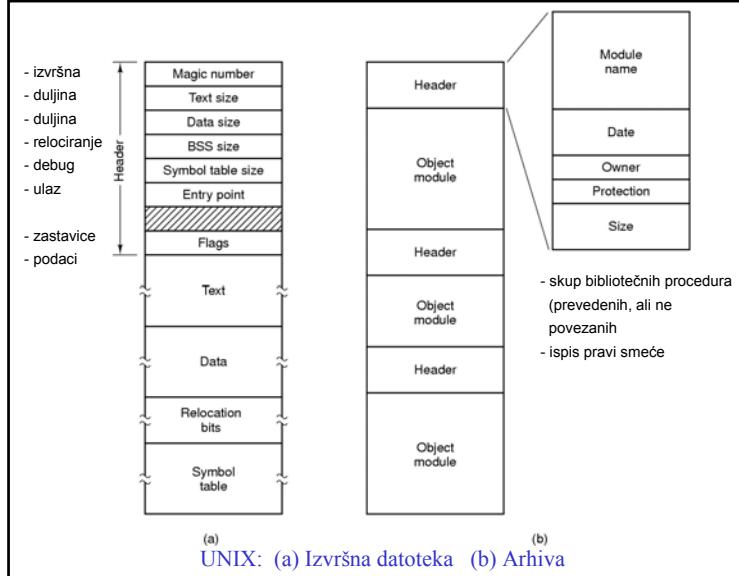
6.1.3. Tipovi datoteka

Regularne datoteke – one koje sadrže korisničke informacije (<- sve 3).
Direktoriji – sustavske datoteke za održavanje strukture datotečnog sustava.
Posebne znakovne datoteke – za modeliranje serijskih U/I uređaja (pisač, terminal)
Posebne blok datoteke – za modeliranje diskova.

Regularne datoteke

- ASCII (linije teksta, prikaz, ispis i uređivanje bilo s kojim editorom, ulaz na izlaz kroz cijev)
- binarne datoteke ("junk" ispis, umutarna struktura poznata programu koji ga koristi)

7



8

6.1.4. Pristup datotekama

- Slijedni pristup
 - čita sve bajtove/slogove od početka
 - ne može preskočiti, može se premotavati
 - u redu ako se radi o magnetskim trakama
- Slučajni pristup
 - bajtovi/slogovi čitaju se bilo kojim redom
 - značajno i za baze podataka
 - Čitanje može biti ...
 - svaka read operacija daje položaj u datoteci gdje treba čitati
 - posebna operacija seek, daje trenutni položaj, a zatim se datoteka može slijedno čitati s tog novog položaja

9

6.1.5. Obilježja datoteka

- svaka datoteka ima ime i podatke, te dodatne podatke ili moguća obilježja:

Attribute	Meaning
Protection	Who can access the file and in what way
Password	Password needed to access the file
Creator	ID of the person who created the file
Owner	Current owner
Read-only flag	0 for read/write; 1 for read only
Hidden flag	0 for normal; 1 for do not display in listings
System flag	0 for normal files; 1 for system file
Archive flag	0 for has been backed up; 1 for needs to be backed up
ASCII/binary flag	0 for ASCII file; 1 for binary file
Random access flag	0 for sequential access only; 1 for random access
Temporary flag	0 for normal; 1 for delete file on process exit
Lock flags	0 for unlocked; nonzero for locked
Record length	Number of bytes in a record
Key position	Offset of the key within each record
Key length	Number of bytes in the key field
Creation time	Date and time the file was created
Time of last access	Date and time the file was last accessed
Time of last change	Date and time the file has last changed
Current size	Number of bytes in the file
Maximum size	Number of bytes the file may grow to

10

6.1.6. Operacije s datotekama

1. Create
2. Delete
3. Open
4. Close
5. Read
6. Write
7. Append
8. Seek
9. Get attributes
10. Set Attributes
11. Rename

11

6.1.7. Primjer programa koji koristi sustavske pozive (1/2) (ne učiti kod napamet)

```
/* File copy program. Error checking and reporting is minimal. */

#include <sys/types.h>           /* include necessary header files */
#include <fcntl.h>
#include <stdlib.h>
#include <unistd.h>

int main(int argc, char *argv[]);      /* ANSI prototype */

#define BUF_SIZE 4096             /* use a buffer size of 4096 bytes */
#define OUTPUT_MODE 0700          /* protection bits for output file */

int main(int argc, char *argv[])
{
    int in_fd, out_fd, rd_count, wt_count;
    char buffer[BUF_SIZE];

    if (argc != 3) exit(1);          /* syntax error if argc is not 3 */

```

12

(2/2)

```
/* Open the input file and create the output file */
in_fd = open(argv[1], O_RDONLY); /* open the source file */
if (in_fd < 0) exit(2); /* if it cannot be opened, exit */
out_fd = creat(argv[2], OUTPUT_MODE); /* create the destination file */
if (out_fd < 0) exit(3); /* if it cannot be created, exit */

/* Copy loop */
while (TRUE) {
    rd_count = read(in_fd, buffer, BUF_SIZE); /* read a block of data */
    if (rd_count <= 0) break; /* if end of file or error, exit loop */
    wt_count = write(out_fd, buffer, rd_count); /* write data */
    if (wt_count <= 0) exit(4); /* wt_count <= 0 is an error */
}

/* Close the files */
close(in_fd);
close(out_fd);
if (rd_count == 0) /* no error on last read */
    exit(0);
else
    exit(5); /* error on last read */
}
```

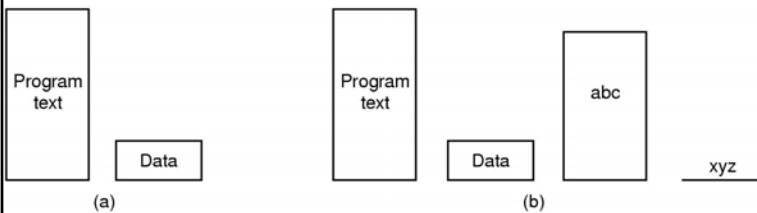
13

6.1.8. Datoteke mapirane u memoriji

- MULTICS: mapiranje datoteka u adresni prostor pokrenutog procesa
- sustavski pozivi: map (daje ime datoteke i virtualnu adresu, koji uzrokuju da OS mapira datoteku u adresni prostor na virtualnu adresu), unmap
- mapiranje datoteka najbolje u sustavima koji podržavaju segmentiranje: svaka datoteka može biti mapirana u vlastiti segment tako da je bajt k u datoteci takođe baji k u segmentu

14

6.1.8. Datoteke mapirane u memoriji



(a) Segmentirani proces prije mapiranja datoteka u njihov adresni prostor

(b) Proces poslije mapiranja

- proces koji ima dva segmenta - tekst i podatke, kopira datoteke. Najprije on mapira izvornu datoteku u segment (abc), a tada kreira prazan segment i mapira ga u odredište (xyz). Proces može kopirati izvorni segment u odredišni uporabom petlje, a bez read write poziva. Kad je sve gotovo – unmap.

postojeća datoteka abc u jedan segment

stvaranje novog segmenta za xyz

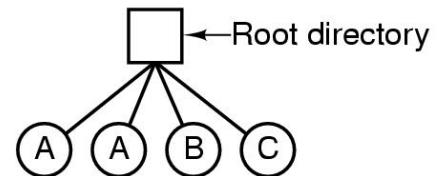
- jednostavnost, ne koristi I/O, ali problem duljine izlazne datoteke, višestrukih pristupa, a datoteka može biti veća od segmenta

15

6.2. Direktoriji (zapravo su isto datoteke)

6.2.1. Jednorazinski sustav direktorija

- sadrži 4 datoteke
- posjeduju ih 3 različita vlasnika A, B i C



- jedan direktorij sadrži sve datoteke (root)

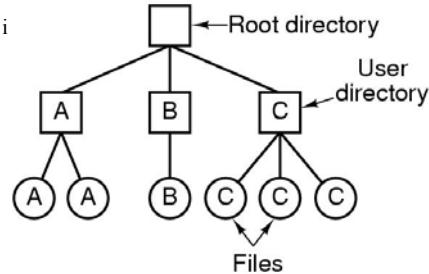
- jednostavno, brzo lociranje datoteke

- problem kod više korisnika koji mogu koristiti ista imena datoteka

16

6.2.2. Dvorazinski sustav direktorija

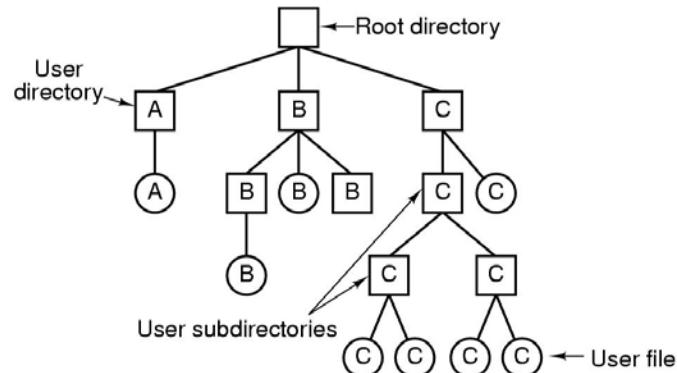
Slova pokazuju *vlasnike* direktorija i



- svaki vlasnik ima vlastiti direktorij
 - nema problema s imenima (višekorisnički ili mrežni sustav)
 - poželjno ili nužno prijavljivanje korisnika
 - pristup direktorijima drugih korisnika (izvođenje programa)

17

6.2.3. Hjерархијски sustav директорија

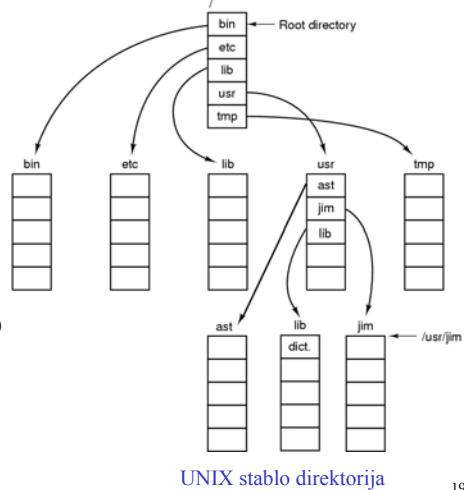


- dvorazinski sustav nepogodan za korisnike s velikim brojem datoteka u skupinama
 - opća hijerarhija (stablo direktorija): svaki korisnik ima potreban broj direktorija

18

6.2.4. Imena putanja (određivanje imena datoteka)

- apsolutno ime: od korijenskog direktorija do datoteke /usr/ast/mailbox (UNIX) \usr\ast\mailbox (Windows)
 - relativno ime: korisnik može načinuti jedan radni (trenutni) direktorij u odnosu na koji se određuju imena putanja datoteka cp /usr/ast/mailbox /usr/ast/mailbox.bak cp mailbox mailbox.bak (isto ako je radni direktorij /usr/ast)



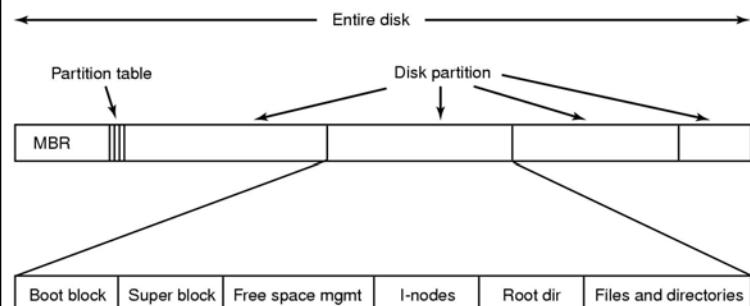
19

6.2.5. Operacije s direktorijima (mapama)

- | | |
|-------------|------------|
| 1. Create | 5. Readdir |
| 2. Delete | 6. Rename |
| 3. Opendir | 7. Link |
| 4. Closedir | 8. Unlink |

20

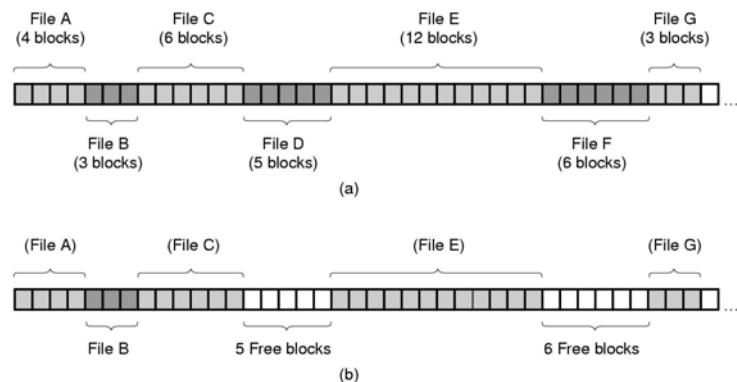
Implementiranje datotečnog sustava (1)



Mogući raspored datotečnog sustava

21

Implementiranje datotečnog sustava (2)

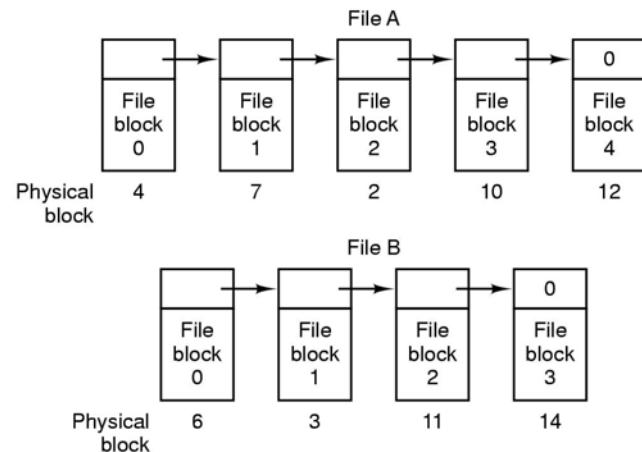


(a) Neprekidno alociranje diskovnog prostora za 7 datoteka

(b) Stanje diska nakon uklanjanja datoteka D i E

22

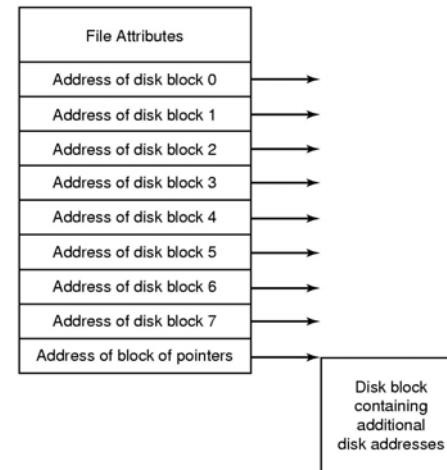
Implementiranje datotečnog sustava (3)



Spremanje datoteke kao povezane liste blokova diska

23

Implementiranje datotečnog sustava (3)



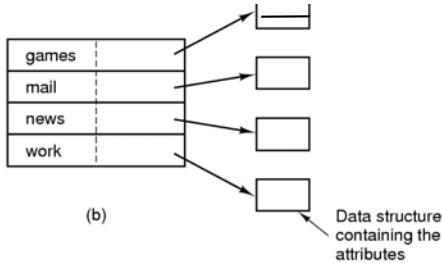
Primjer i-čvora

24

Implementiranje direktorija (1)

games	attributes
mail	attributes
news	attributes
work	attributes

(a)



(b)

Data structure
containing the
attributes

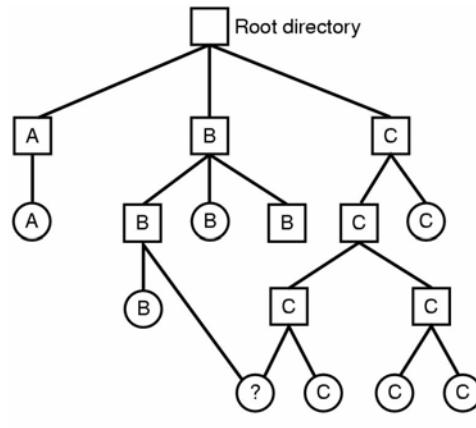
(a) Jednostavan direktorij

- dijelovi jednake duljine
- adrese na disku i atributi su u direktoriju

(b) Direktorij u kojem se svaki dio referira na tzv. i-node tablicu

25

Dijeljene datoteke (1)

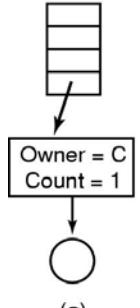


Datotečni sustav koji sadrži dijeljene datoteke

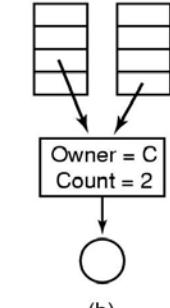
26

Dijeljene datoteke (2)

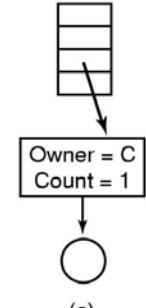
C's directory



B's directory C's directory



B's directory



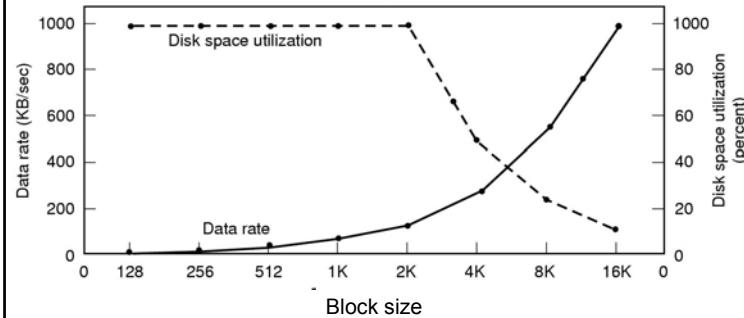
(a) Prije povezivanja

(b) Nakon što je načinjeno povezivanje

(c) Nakon što izvorni vlasnik udalji datoteku

27

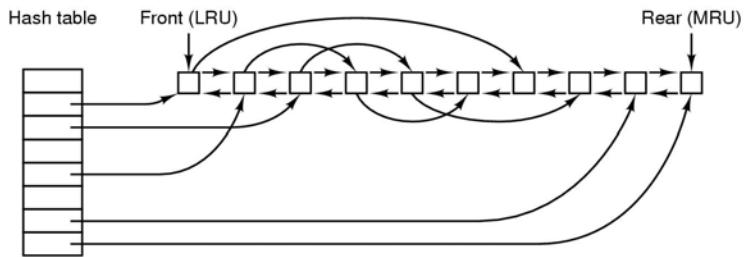
Rukovanje prostorom diska (1)



- Sve datoteke su 2KB
- Rukovanje može biti najčešće povezanim listama i bitmapama

28

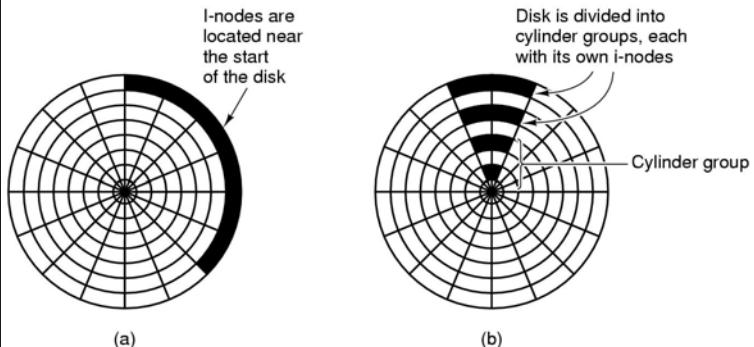
Performanse datotečnog sustava (1)



The block cache data structures

29

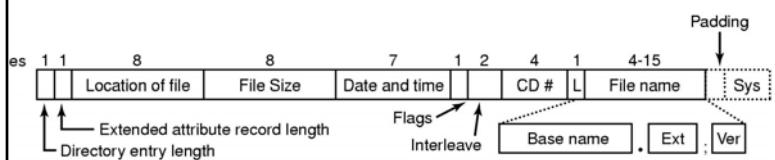
Performanse datotečnog sustava (2)



- I-čvorovi smješteni na početku diska
- Disk podijeljen u skupine cilindara
 - svaki ima vlastite blokove i i-čvorove

30

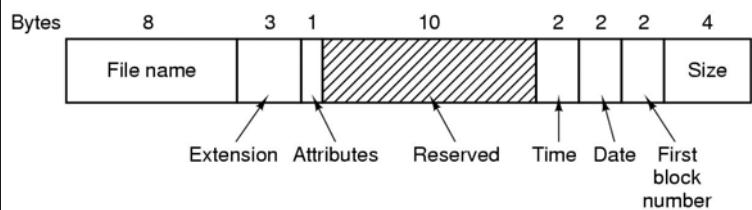
Primjeri datotečnih sustava CD-ROM datotečni sustav



ISO 9660 directory entry

31

MS-DOS datotečni sustav (1)



MS-DOS directory entry

32

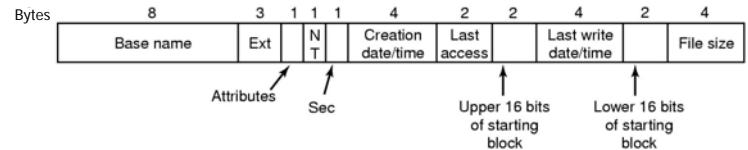
MS-DOS datotečni sustav (1)

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

- Maksimalna particija blokova različitih duljina
- Prazna polja označavaju zabranjene kombinacije

33

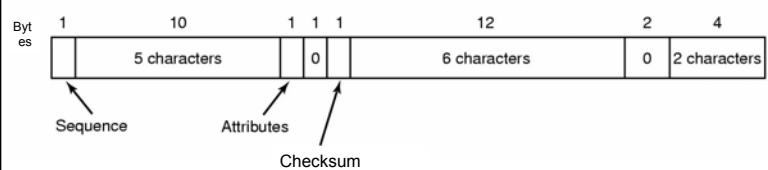
Windows 98 datotečni sustav (1)



The extended MOS-DOS directory entry used in Windows 98

34

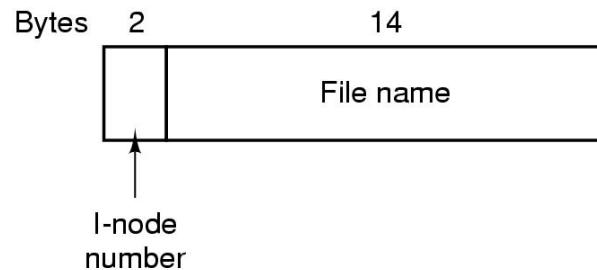
Windows 98 datotečni sustav (2)



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

35

UNIX V7 datotečni sustav (1)



A UNIX V7 directory entry

36

