

**UNIVERSITATEA POLITEHNICA BUCURESTI**

**FACULTATEA DE ELECTRONICA, TELECOMUNICATII SI TEHNOLOGIA  
INFORMATIEI**

**Proiect SO**

**Sisteme de fisiere**

**Studenti:**

**Oprea Robert – Georgian 433A**

**Visan Valentin 431A**

# Sisteme de fişiere

## Sisteme de fişiere – privire de ansamblu

- de ce sunt sistemele de fişiere necesare?
  - nevoia de a stoca un volum mare de date
  - mentinerea persistenţei informaţiei
  - accesul concurenţial la informaţie
- unitatea fundamentală de stocare a informaţiei este fişierul
- sistemul de fişiere este o metodă de organizare şi stocare a fişierelor pentru un acces rapid şi eficient
- sistemele de fişiere necesită un dispozitiv de stocare (hard disk, CD-ROM, floppy disk)

## Sisteme de fişiere - componente

- un sistem de operare poate fi evaluat din două perspective:
  - din perspectiva utilizatorului – logical file system
  - din perspectiva sistemului de operare
- important pentru utilizator este cum îi apare sistemul de fişiere, ce reprezintă un fişier, cum se denumeşte un fişier, ce operaţii se permit asupra fişierului, etc.
- pentru sistemul de operare, modul de depozitare a informaţiei pe disc este foarte important, de asemenea, la fel de important fiind modul de folosire a unor algoritmi eficienţi pentru acoperirea blocurilor şi sectoarelor de pe dispozitivul de stocare

## **Perspectiva utilizatorului**

- din punctul de vedere al utilizatorului modul de reprezentare al sistemului de fisiere este aspectul cel mai vizibil al sistemului de operare
- doua componente sunt esentiale din punct de vedere al utilizatorului: fisierele si directoarele
- unele sisteme de fisiere contin si partitii –pentru separarea fizica/logica a unui numar mare de directoare
- intr-un mediu multiuser este necesar conceptul de drepturi de acces la fisiere (file protection)

## **Fisiere**

- unitatea logica de stocare
- abstractizeaza proprietatile fizice ale dispozitivului destocare
- pot fi programe sau date (numerice, alfanumerice, binare)
- in general, un fisier este o succesiune de octeti, de linii sau de inregistrari
- unui fisier ii este asociat un nume pentru a putea fi usor referit
- unele sisteme sunt case-sensitive (UNIX), altele caseinsensitive
- sistemele DOS impun folosirea unei extensii pentru numele fisierului
- folosirea unei extensii poate fi impusa de un program (de exemplu un fisier C)

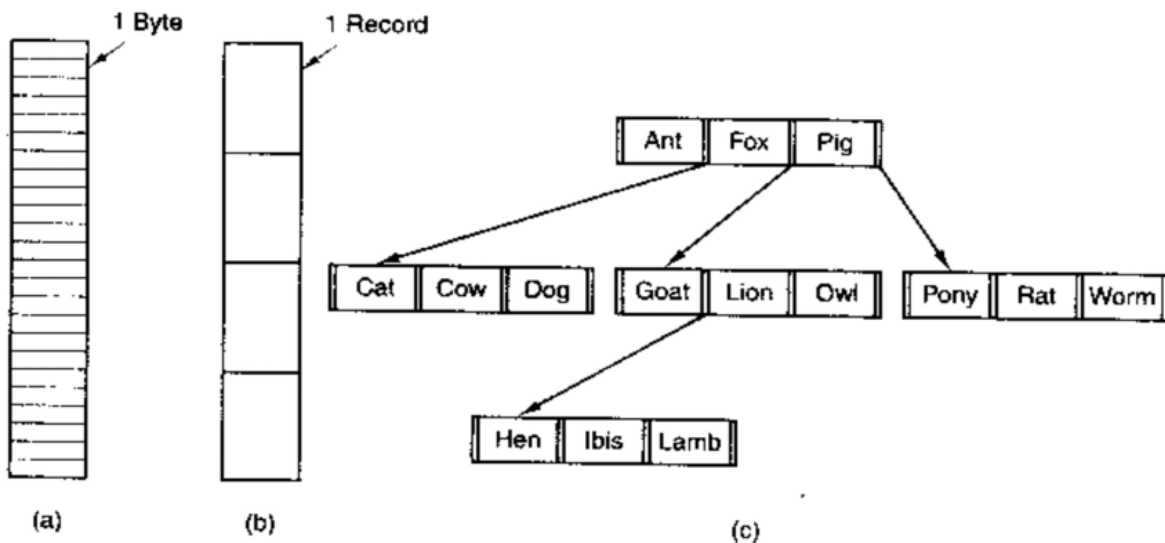
## **Extensia unui fisier**

- extensia unui fisier poate fi utilizata pentru descrierea simpla a tipului unui fisier:
  - exe, com, bin – fisier executabil
  - obj, o – fisier obiect

- c, pas, cpp, f77, asm – fișiere cod sursa
- bat, sh – script-uri shell (comenzi catre interpretor)
- acr, zip, tar, rar – arhive

## Structura unui fișier

- structura unui fișier:
  - byte stream – sir de octeti fara organizare speciala
  - record – format din inregistrari
  - organizare arborescenta
- folosirea unui sir de octeti asigura flexibilitate maxima (folosit in Windows, Unix)
- organizarea arborescenta permite localizarea rapida a informatiei



**Figure 6-2.** Three kinds of files. (a) Byte sequence. (b) Record sequence. (c) Tree.

## Tipuri de fișiere

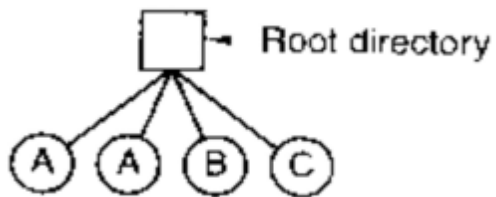
- tipuri de fișiere
  - obisnuite (regular) – fișiere binare sau ASCII
  - directoare – pentru mentinerea structurilor sistemului de fișiere
  - fișiere speciale (UNIX)
- fișiere speciale tip caracter: pentru modelarea dispozitivelor de I/E seriale (terminale, imprimante): /dev/lp0; /dev/ttyS0
- fișiere speciale tip bloc: /dev/hda, /dev/hdb, /dev/fd0
- fișierele binare au asociate un magic number pentru identificarea usoara

## Operații asupra fișierelor

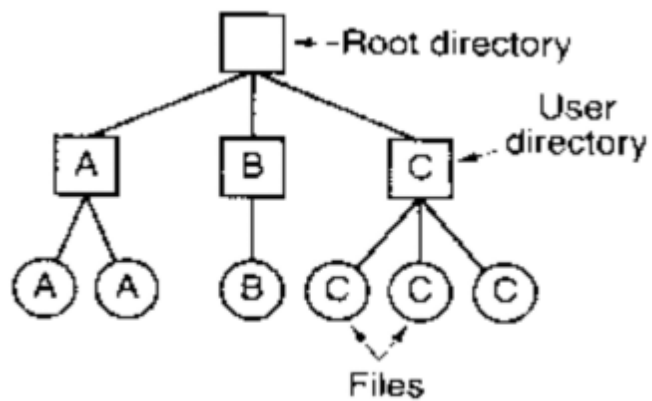
- tipuri de acces
  - secvential
  - aleator
- operatii posibile asupra unui fisier:
  - creare
  - stergere
  - deschidere
  - inchidere
  - citire
  - scriere
  - adaugare (append)
  - cautare (seek)
  - redenumire

## Directoare

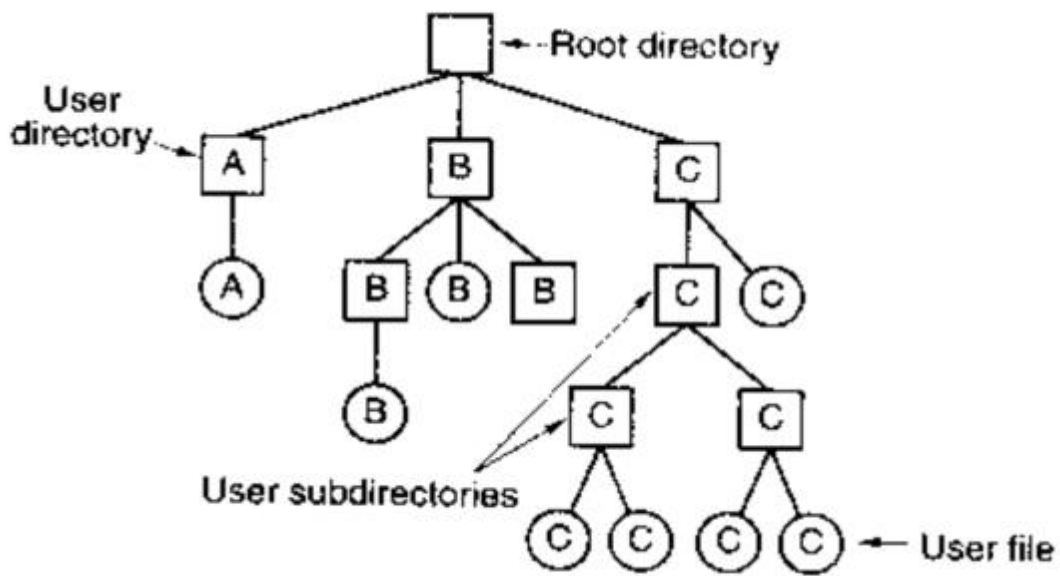
- directories, folders – un tip special de fisiere
- single directory systems
  - un singur director care contine toate fisierele (root directory)
  - problema este ca, intr-un mediu multiuser, mai multi utilizatori pot da acelasi nume la diverse fisiere



- two-level directory systems
  - se atribuie fiecarui utilizator un director privat
  - se elimina problema interferentei numelor
  - nesatisfacator pentru utilizatori cu numar mare de fisiere



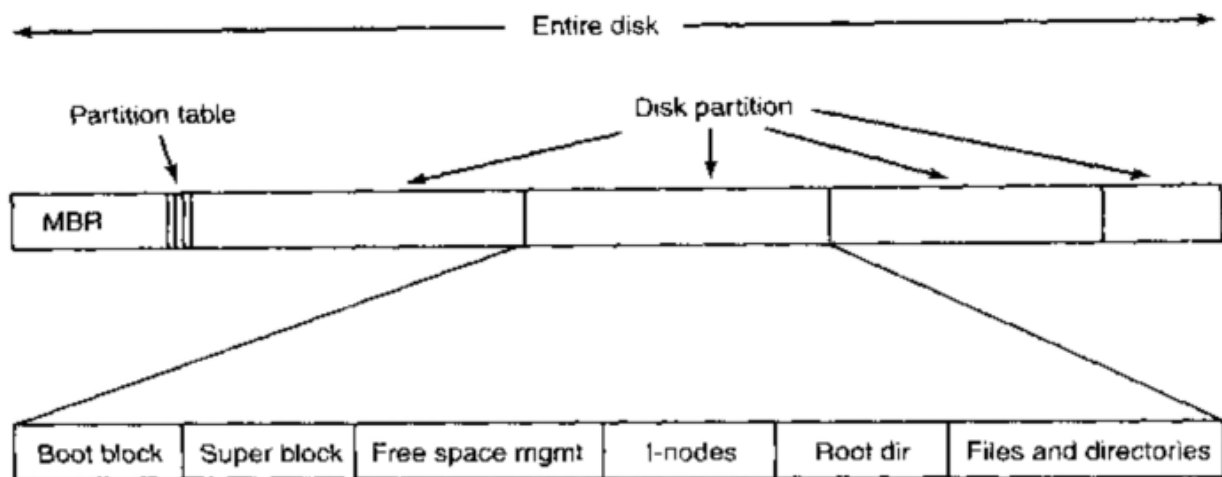
- structura ierarhica a sistemului de fisiere



- path name – cai de nume
  - relative: simple.c; cfiles/simple.c
  - absolute: /home/student/cfiles/simple.c
- tipuri de operatii pe directoare
  - creare
  - stergere
  - deschidere
  - inchidere
  - citirea informatiilor din director
  - redenumire
  - link (obtinerea unui link)

## Sisteme de fisiere – aspect general

- sistemele de fisiere sunt stocate pe disc
- sectorul 0 contine MBR (Master Boot Record); spre sfarsit se afla tabela de partitii
- se localizeaza partitia activa, blocul de boot si se executa
- toate sistemele de fisiere incep cu sectorul de boot, dar in rest difera;
- de obicei contine:
  - superbloc
  - bloc de control al spatiului liber
  - directorul radacina
  - fisierele si directoarele



**Figure 6-11.** A possible file system layout.



## **Montarea unui sistem de fisiere**

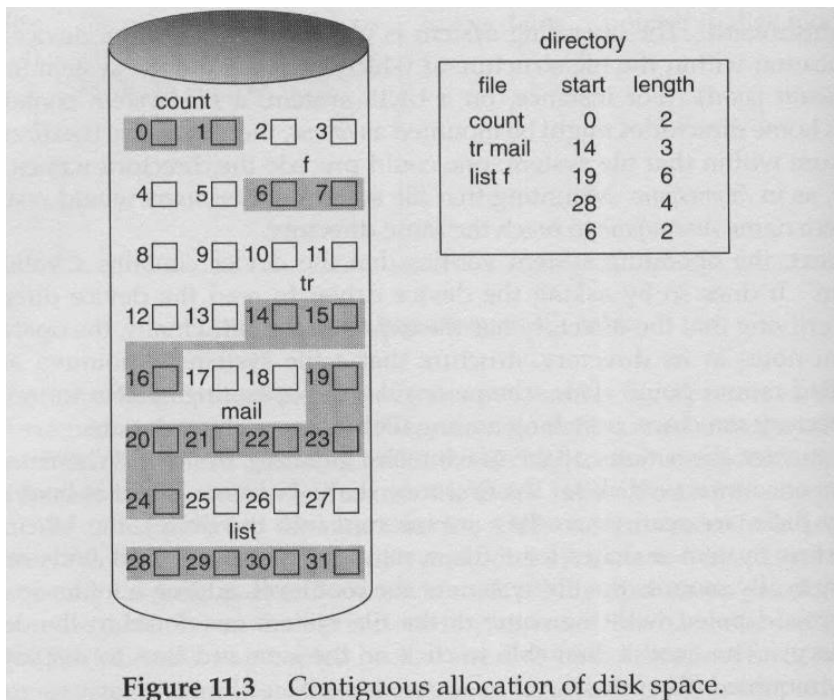
- un fisier trebuie sa fie deschis pentru a fi utilizat
- un sistem de fisiere trebuie montat inainte de utilizare
- montare = sistemul de operare primeste numele unui dispozitiv si un punct in structura de fisiere unde se va atasa sistemul de fisiere
- sistemul de operare verifica existenta unui sistem de fisiere valid si se realizeaza montarea
- la Windows, hard disk-urile sunt detectate la boot time si sunt montate respectiv in C:, D:, etc.

## **Metode de alocare**

- specifica modul de stocare a informatiei pe disc (in cadrul unui sistem de fisiere)
- scopuri:
  - utilizare eficienta a spatiului disponibil
  - accesul rapid la fisiere
- trei metode principale de alocare a spatiului:
  - alocare contigua (contiguous allocation)
  - alocare cu liste (linked allocation)
  - alocare indexata (indexed allocation)

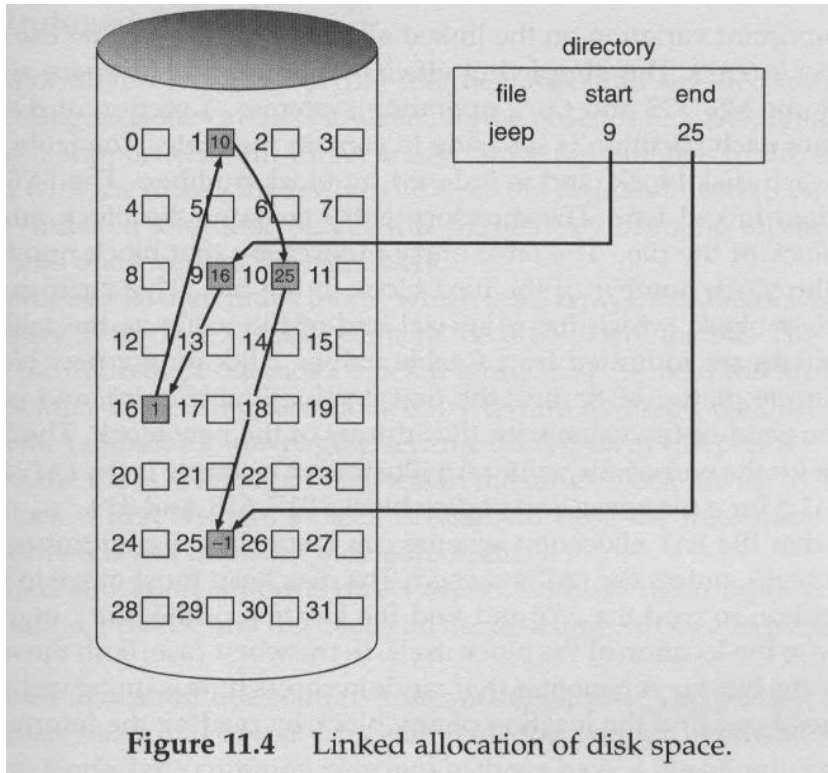
## Alocare contigua

- un fisier ocupa o secventa contigua de blocuri pe disc
- probleme de fragmentare



## Alocarea cu liste inlantuite

- elimina problemele de fragmentare
- este potrivita pentru acces secvential; pentru accesul aleator se fac multe operatii



## Alocarea indexata – inode

- pointer-ii catre toate blocurile ce contin un fisier sunt stranse intr-o structura specializata: index block

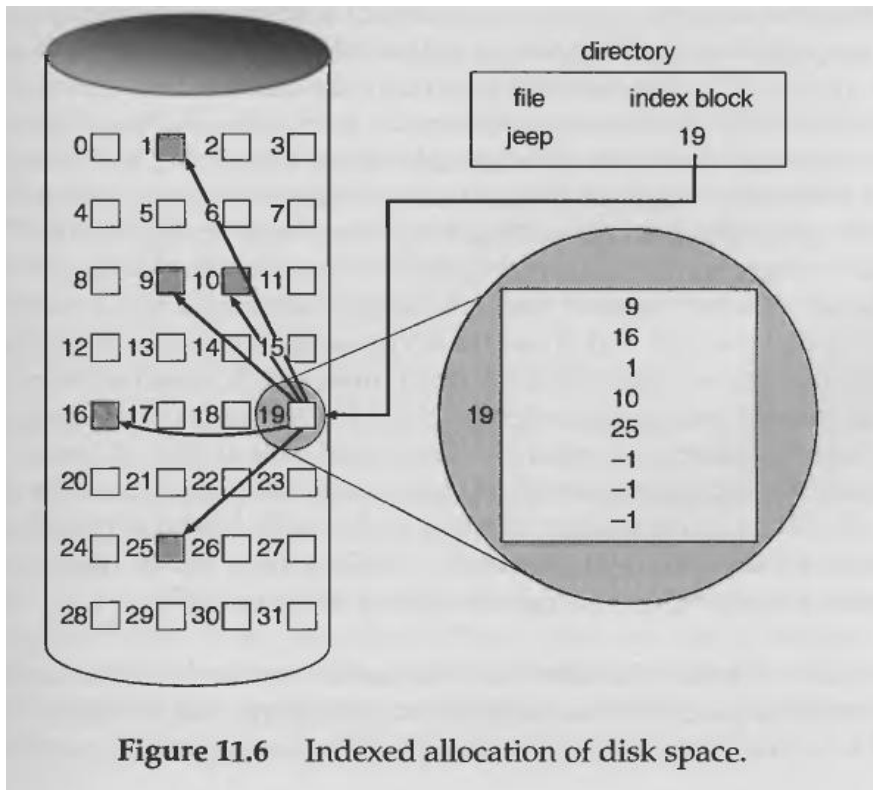
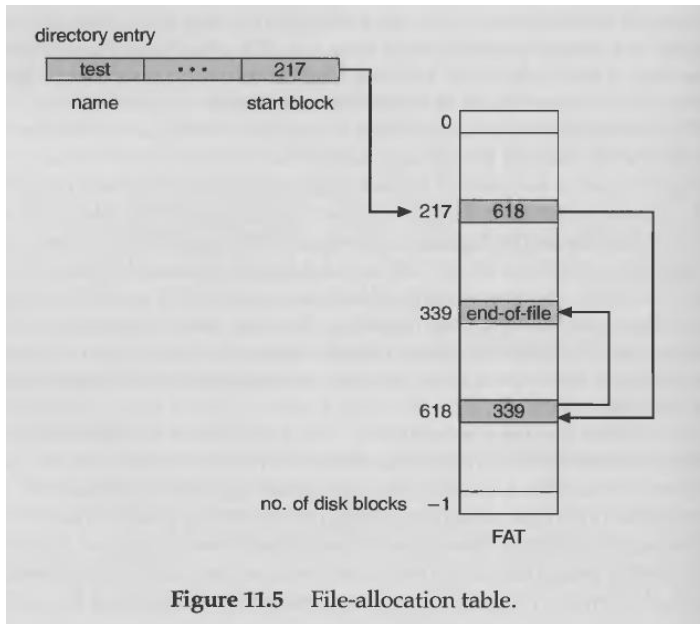


Figure 11.6 Indexed allocation of disk space.

## Tipuri de sisteme de fisiere

### FAT – File Allocation Table

- Folosit in MS-DOS, Windows '95, Windows '98
- contine:
  - sectorul de boot
  - FAT region
  - Root Directory Region
  - Data Region
- o tabela FAT este o lista de intrari care se mapeaza pe fiecare cluster si contine:
  - adresa urmatorului cluster
  - un indicator special End Of File
  - un caracter special pentru marcarea “bad blocks”
  - un caracter special care sa indice un cluster rezervat
  - zero ca sa indice un cluster neutilizat



## NTFS – New Technology File System

- Windows 2000, Windows XP, Windows 2003
- inlocuire pentru FAT
- s-au adaugat liste de control al accesului (drepturi de acces) si jurnalizare
- imbunatatiri:
  - volume mount points (specificarea unei locatii de montare, in locul crearii unui noi drive C:, D:)
  - file compression
  - encrypting file system
  - cote (quotas)

## Ext – Extended File System

- sistemul nativ Linux
- ext, ext2, ext3
- ext3 – folosit actualmente de multe din distributiile Linux – ext2 + facilitati de jurnalizare
- sistem cu alocare indexata (i-node); ext a fost bazat pe sistemul de fisiere Minix
- fsck (file system check) – utilitarul pentru verificarea consistentei unui sistem de fisiere tip

Unix

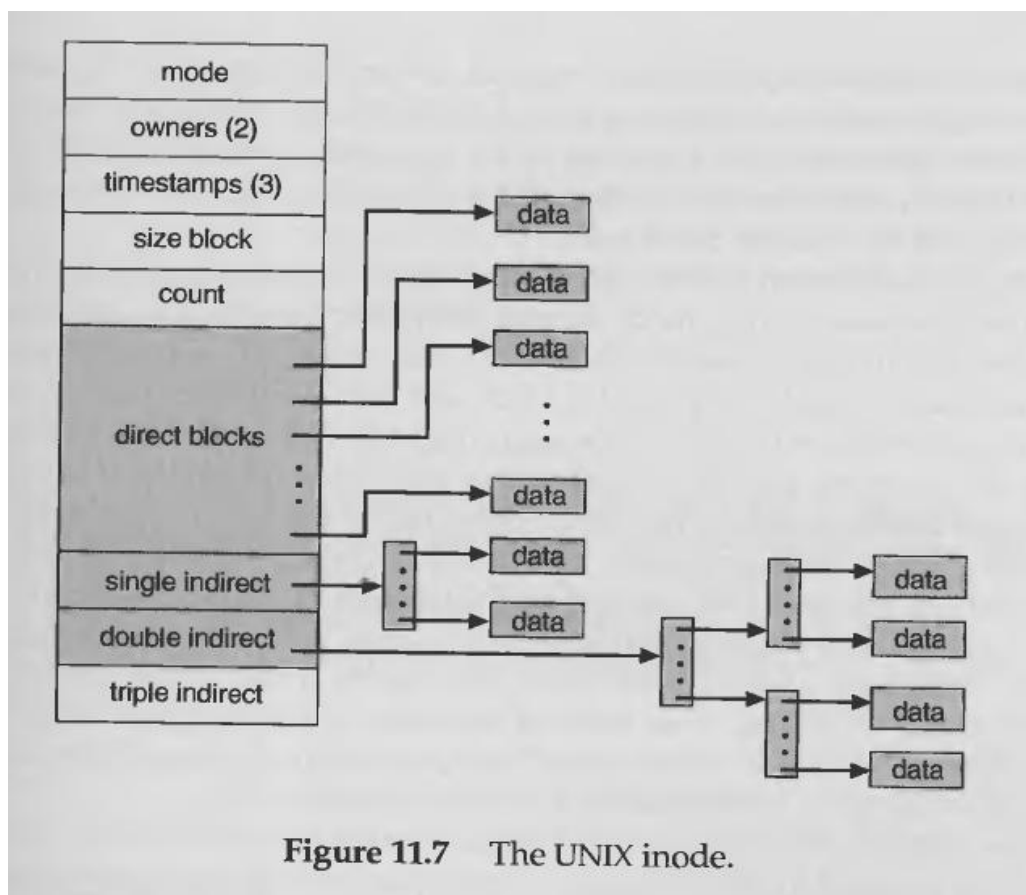


Figure 11.7 The UNIX inode.

## **ReiserFS, Reiser4**

- sistem de fisiere general-purpose
- inclus in kernel-ul Linux
- sistemul de fisiere implicit pe multe distributii Linux (Slackware, SuSE, Xandros, Linspire)
- jurnalizare, online resizing, tail packing (schema de reducere a fragmentarii)
- pentru fisiere mai mici de 4K, este mai rapid decat ext3 de 10-15 ori
- Reiser4 este versiunea actuala de dezvoltare (nu a fost inclusa inca in kernel-ul Linux)
- aduce imbunatatiri pentru lucrul cu fisiere cu format specific

## **Fiabilitatea unui sistem de fisiere**

- posibilitatea recuperarii din eroare
- distrugerea unui sistem de fisiere este adesea mai grava decat distrugerea unui calculator
- solutii?
  - Copii de siguranta (backups)
  - pastrarea consistentei (modificarile asupra blocurilor nu se fac instantaneu pe disc)
- scandisk si fsck – verifica consistenta unui sistem de fisiere si, folosindu-se de redundanta acestuia, il aduc la o stare valida

Bibliografie: [www.wikipedia.com](http://www.wikipedia.com)

“Structured Computer Organization”, Andrew S. Tanenbaum

“Operating Systems – Design an Implementation”, Andrew S. Tanenbaum