

## Capitolul 2

# SISTEMUL DE FIȘIERE

### 1.1. STRUCTURA DE FIȘIERE

Una dintre cele mai importante caracteristici Linux este modul în care tratează fișierele. În Linux **totul este un fișier** (aproape totul), programele utilizează porturile seriale, imprimantele și alte dispozitive în același mod în care utilizează fișierele de pe discuri.

Sistemul de operare Linux utilizează trei tipuri de fișiere:

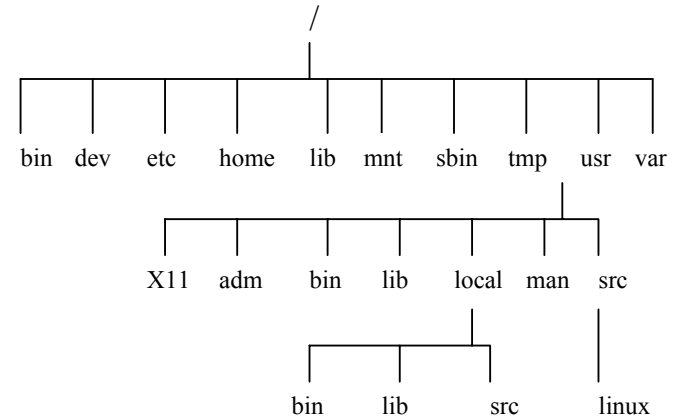
- Fișiere obișnuite – sunt doar colecții de octeți asupra cărora sistemul nu impune nici o structură particulară. Acestea sunt utilizate pentru fișierele text, fișierele de date pentru programe și pentru fișierele executabile ale programelor.
- Cataloage – un catalog este un mecanism care permite fișierelor să fie grupate împreună, reprezintă un container în care pot fi plasate alte cataloage și fișiere. De fapt catalogul conține doar numele fișierelor și informații despre locația acestora. Deoarece un catalog poate cobține subcataloage avem de-a face cu o structură ierarhică.
- Dispozitive – această categorie este alcătuită din anumite tipuri de fișiere, cum ar fi cele asociate comunicației între procese sau cele asociate comunicației cu diverse dispozitive hardware ca terminale, CD-ROM, modem, etc.

Toate aceste fișiere sunt organizate într-o structură ierarhică arborescentă, având ca și vârf un singur catalog care poartă numele de “rădăcină”, el fiind reprezentat prin “/”. Acest catalog “rădăcină” conține o serie de alte cataloage ce au destinații speciale. Aceste destinații sunt doar convenții și nu sunt obligatorii.

Semnificația acestor cataloage:

/bin	executabilele comenzilor
/dev	fișierele speciale
/etc	fișierele de configurare
/home	cataloagele utilizatorilor
/lib	bibliotecile standard

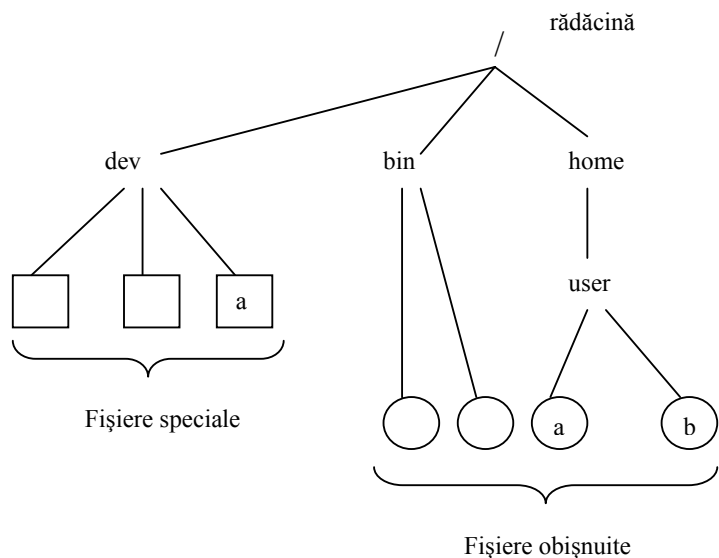
/mnt	
/sbin	comezi de administrare
/tmp	fișiere temporare
/usr	date disponibile utilizatorilor
/var	fișiere mari în cazuri speciale



Este posibil ca două fișiere să aibă același nume atâta timp cât sunt în cataloage diferite, referirea acestor fișiere făcându-se prin intermediul numelui de cale. Acesta poate fi:

- *absolut* ca punct de referință se ia catalogul “rădăcină”, ex:  
/home/user/a  
/home/dev/a
- *relativ* punctul de referință este reprezentat de un catalog arbitrar:  
/user/a  
/user/b

La fiecare schimbare a catalogului noul catalog devine *catalogul curent de lucru*. Acest lucru este foarte important deoarece pentru fiecare program executat în Linux va rezulta un proces care va avea ca parte a stării lui interne și catalogul curent de lucru. Acesta fiind catalogul utilizat ulterior pentru referiri bazate pe calea relativă. Dacă procesul nu face aranjamente specifice pentru a utiliza un alt catalog de lucru, atunci implicit va utiliza catalogul în care a fost utilizatorul când programul a fost pornit.

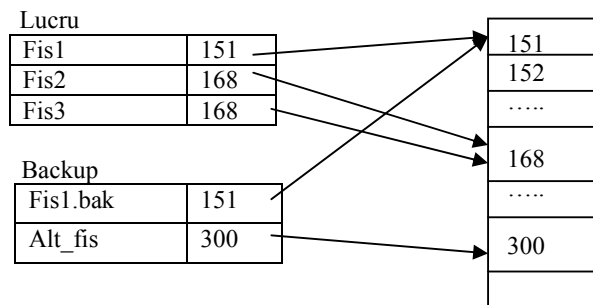


Fiecărui fișier, stocat pe disc, indiferent de tipul său i se asociază un număr numit *inode* care reprezintă de fapt un index al unei intrări într-un șir stocat pe disc. Fiecare element al șirului este un *inode* care stochează în formații specifice unui anumit fișier (când a fost creat, cine îi este proprietarul și unde sunt stocate blocurile de date pentru fișierul respectiv). Un catalog conține pe lângă numele fișierului și numărul de *inode* asociat acestuia. Fiecare pereche nume – inode dintr-un catalog poartă numele de legătură sau *link*.

În acest fel este posibil ca același nume de *inode* să apară în unul sau mai multe *link*-uri. Deci un fișier poate avea unul sau mai multe nume de cale valide.

Cataloage

Șirul de *inode*



## 1.2. COMENZI SPECIFICE LUCRULUI CU FIȘIERE

Imediat după intrarea în sistem, implicit utilizatorul va fi plasat într-un catalog asociat cu numele de *login* numit *home*. Este catalogul care va fi catalogul curent de lucru inițial.

Cel mai simplu mod de a afla unde se află catalogul *home* situat în ierarhia de cataloage este prin comanda:

```
pwd – present working directory
parametrii – fără
$ pwd
/home/user
rezultat – afișarea catalogului curent de lucru
```

Pentru schimbarea catalogului curent de lucru este disponibilă comanda:

```
cd – change directory
parametrii – catalogul destinație
– implicit catalogul home al utilizatorului
$ cd /
rezultat – catalogul curent de lucru este catalogul rădăcină
```

Pentru a obține o listă a numelor fișierelor și cataloagelor se utilizează comanda: *ls*

```
bash$ ls -l /usr/lib/X11/Motif
total 392
-r--r--r-- 1 root root 142509 Nov 25 1993 CLOSEDBUGS
drwxr-xr-x 2 root root 1024 Nov 25 1993 DragAndDrop
-r--r--r-- 1 root root 216209 Nov 25 1993 OPENBUGS
-r--r--r-- 1 root root 12278 Nov 25 1993 README-1,2,3
drwxr-xr-x 2 root root 1024 Nov 25 1993 clipboard
Parametrii – l afișare extinsă
$ ls
```

**De studiat!** Utilizând comenzile mai sus prezentate să se parcurgă și să se afișeze conținutul catalogului *home*.

Comenzi specifice cataloagelor:

```
mkdir – creează noi cataloage.
rmdir – șterge un catalog, dacă nu conține nici un fișier.
```

**De studiat!** Să se creeze un catalog cu numele *work* în catalogul *home* al fiecărui utilizator.

Comanda de copiere în Linux este **cp**. Acceptă ca și parametrii calea a două fișiere obișnuite realizând o copie a primului în al doilea. O altă formă este cea în care al doilea parametru este un catalog.

```
$cp fis1 fis2
```

rezultat: fis1 este copiat în fis2

```
$cp fis2 /home/user
```

rezultat: fis2 este copiat în /home/user

În cazul în care se dorește să se șteargă un catalog trebuie ca în primul rând să se șteargă fișierele obișnuite conținute în acel catalog. Pentru ștergerea acestor fișiere folosește comanda **rm**.

```
$rm fis1
```

rezultat: șterge fis1

**Atenție!** În Linux ștergerea fișierelor nu este reversibilă, de aceea se recomandă utilizarea cu atenție a utilitatelor de ștergere.

Dacă se dorește mutarea unuia sau mai multor fișiere se poate utiliza utilitarul **mv** efectul este același ca și în cazul utilizării combinate a comenzilor cp și rm. Comanda **mv** se poate folosi și pentru modificarea numelui fișierelor primite ca și argumente.

**De studiat!** Pentru toate comenzile prezentate pâna acum să se utilizeze comanda man pentru a se studia posibilele opțiuni.

**De studiat!** – pagina de manual pentru: mv, ln

O categorie aparte de comenzi pentru lucrul cu fișierele o reprezintă cele destinate prelucrării acestora. Dintre acestea cele mai importante sunt:

**Cat** – este prescurtarea pentru *concatenate* și este folosită pentru concatenarea a două sau mai multe fișiere.

```
$cat fis1 fis2
```

rezultat fis1 este adăugat la sfârșitul fis2

*Cat* se poate folosi și fără argumente :

```
$cat
```

*Acest text este doar un exemplu*

```
Ctrl-d
```

Rezultatul este afișarea textului pe ecran pentru că implicit această comanda primește ca și ieșire standard ecranul iar ca și intrare standard tastatura.

**De studiat!** Să se afișeze pe ecran utilizând *cat* conținutul fișierului fis1 utilizat în temele anterioare.

Comanda *cat* se poate utiliza în mod eficient numai pentru afișarea unor fișiere de mici dimensiuni care să încapă pe ecran. Pentru cele de dimensiuni mai mari se folosește **more**.

**De studiat!** – pagina de manual pentru: head, tail

**De studiat!** Să se afișeze numărul de cuvinte din fișierul *etc/passwd* (indicație folosiți *wc*)

**De studiat!** Să se studieze pagina de manual pentru următoarele comenzi: date, echo.