

## 8 SISTEMELE DE OPERARE ACTUALE

Sistemele de operare se clasifică în două categorii:

- a) **sisteme proprietare**, care sunt sisteme specifice unei mașini sau unei game de mașini ale unui anumit constructor, ca de exemplu VMS, elaborat de firma DEC pentru gama VAX, MS-DOS, pentru microcalculatoare bazate pe microprocesoare Intel, VM, pentru calculatoare IBM, MACOS pentru Macintosh al firmei Apple etc.
- b) **sisteme deschise**, care sunt sisteme portabile pe diverse mașini. Sistemul Unix este exemplul tipic al unui sistem deschis, el a fost implementat pe mașini dintre cele mai diverse, de la microcalculatoare și până la supercalculatoare.

Sistemele proprietare au tendința să dispară în beneficiul sistemelor deschise standard.

O nouă generație de sisteme de operare deschise se dezvoltă la ora actuală, (ca de exemplu, sistemul Windows NT al firmei MicroSoft) sisteme deschise care funcționează în regim de multiprogramare, multiutilizator, cu o interfață grafică, și care integrează multiple posibilități de lucru în rețea.

### 8.1 Sistemul DOS

**DOS** [Disk Operating System] este un sistem de operare care cuprinde programe de interfață între utilizator și sistemul de calcul stocate pe disc magnetic.

Structura sistemului de operare DOS este următoarea:

- a) **Componenta ROM-BIOS** este livrată de către firma producătoare a sistemului de calcul și are rolul de a egaliza toate diferențele constructive ale sistemelor de calcul față de convențiile DOS, deci calculatoarele sunt compatibile. ROM-BIOS oferă rutine de întrerupere prin care se realizează legătura cu toate unitățile periferice conectate la sistem;
- b) **Componenta DISK-BIOS**, este independentă de hardware-ul sistemului de calcul și ea extinde funcțiile ROM-BIOS. Această componentă este materializată printr-un fișier disc care, în funcție de versiunea sistemului de operare poate avea unul dintre numele: IBMBIO.COM, BIO.COM, IO.SYS, și este rezidentă pe discheta sau discul Winchester de pe care se încarcă sistemul. Funcțiile BIOS care vizează a) și b) au în vedere următoarele sarcini:
  - încărcare sistem de operare;
  - determinarea și testarea echipamentelor periferice;
  - testarea memoriei RAM;

- funcții de bază asupra ecranului video;
- funcții elementare de citire a tastaturii;
- funcții elementare de lucru cu dischetele sau cu hard-discul;
- funcții de lucru cu caseta magnetică;
- funcții de lucru cu liniile de comunicație asincronă (imprimantă, scanner, plotter etc.);

Pentru lucrul cu discul, BIOS operează cu cilindri, piste și sectoare.

c) **Componenta** BDOS extinde funcțiile BIOS la un nivel mai înalt, în special către lucrul cu discurile. Această componentă este materializată printr-un fișier disc care, în funcție de versiunea sistemului de operare poate avea unul dintre numele: IBMDOS.COM, DOS.COM, MSDOS.SYS, și este rezidentă pe discheta sau discul Winchester de pe care se încarcă sistemul. Cele mai importante sarcini BDOS sunt următoarele:

- gestiunea memoriei: alocarea și eliberarea spațiului de memorie necesar programelor;
- gestiunea proceselor (programelor);
- execuția operațiilor de intrare/ieșire cu periferice de tip caracter;
- tratarea fișierelor disc: creare, deschidere, închidere, accesare, ștergere;
- partajarea fișierelor între mai multe procese active la un moment dat;
- gestiunea structurii arborescente a fișierelor și manipularea directoarelor ;
- gestiunea rețelelor de calculatoare;
- modificări ale sistemului de întrerupere;
- gestiunea ceasului intern și a calendarului.

d) **Componenta** COMMAND are ca sarcină principală preluarea comenzilor formulate de utilizator de la tastatură și lansarea lor în execuție. Această componentă este materializată prin fișierul disc COMMAND.COM. Comenzile DOS sunt de două tipuri:

- **comenzi interne**, înglobate în COMMAND și încărcate permanent în memorie. De regulă, acestea sunt comenzi mai simple și de primă utilitate: rezumate de disc, copieri de fișiere etc.
- **comenzi externe**, rezidente pe disc sub forma unor programe executabile sau sub forma unor fișiere de comenzi (fișiere text în care sunt apelate alte comenzi DOS). De regulă, comenzile externe sunt mai complexe decât cele interne.

În timpul introducerii comenzilor, DOS recunoaște o serie de funcții de control, din care amintim câteva:

- <CTRL/C> sau <CTRL/Break>: abandonează linia de comandă;
- <F3>: copiază pe ecran ultima linie introdusă;
- <CTRL/P>: trimite în ecou ieșirea pe ecran și imprimantă;
- <CTRL/N>: anulează efectul lui <CTRL/P>;
- <CTRL/S>: oprește temporar defilarea ecranului, repornirea se realizează prin apăsarea oricărei taste cu excepția <CTRL/C>;
- <F6> sau <CTRL/Z>: inserează marcator de sfârșit fișier;
- <F1>: copiază un caracter din șablon și îl afișează;
- <F2>: copiază toate caracterele care preced un caracter specificat.

Generarea sistemului de operare DOS pe o dischetă sau pe un hard-disc se realizează prin intermediul unor comenzi standard, ștergând toate informațiile vechi de pe discul respectiv. Etapele generării unui disc DOS sunt următoarele:

- a) **Partajarea hard-discului:** împărțirea în una până la patru zone contigue numite **partiții**. În acest fel, în loc de un singur disc mare, DOS “vede” mai multe discuri mici. Operația de partajare se realizează cu ajutorul programului FDISK, care printr-un dialog cu utilizatorul realizează partiționarea la dimensiunile dorite;
- b) **Formatarea discului.** Comanda DOS FORMAT realizează formatarea discului (împărțire în cilindri, piste, sectoare) și verificarea sectoarelor sale. Tot prin această comandă se crează **discul sistem DOS** (opțiunea /S), iar ca efect, pe discul respectiv se realizează următoarele:

- în sectorul 0 se pune programul **bootstrap** de încărcare a sistemului de operare;
- se depun pe disc fișierele IBMBIO.COM, IBMDOS.COM și COMMAND.COM;
- în restul sectoarelor se pune valoarea 0.

Dacă s-a procedat la partajarea hard-discului este necesară formatarea tuturor unităților de disc logic care au ca suport fizic discul partajat. Este suficientă formatarea cu opțiunea “/S” numai a uneia dintre partiții (cea care este declarată activă prin FDISK);

- c) **Crearea fișierului de configurare.** La fiecare lansare a sistemului se execută o operație de **configurare** prin care se fixează valorile unor parametri sistem, cum ar fi:
  - numărul fișierelor deschise simultan;
  - numărul zonelor tampon pentru operațiile de intrare/ieșire;
  - numărul și dimensiunea stivelor;

- condiția de sesizare a tastării <CTRL/Break>;
- convențiile de țară, timp, monedă;
- precizarea unor noi drivere de intrare/ieșire;

Acești parametri sunt precizați prin intermediul unui fișier text numit CONFIG.SYS. Dacă fișierul nu este prezent pe disc, atunci DOS atribuie acestor parametri niște valori implicite, care depind de tipul sistemului de calcul. De obicei, conținutul acestui fișier este stabilit la generare în funcție de configurația hardware și de aplicațiile care vor fi rulate pe sistemul respectiv. Fișierul se crează simplu, cu ajutorul unui editor de texte, eventual el poate fi pregătit dinainte, urmând a fi copiat pe noul disc;

- d) **Crearea fișierului de comenzi inițiale.** În momentul lansării sistemului, după fixarea parametrilor de configurare și după încărcarea interpretorului de comenzi, DOS caută pe disc un fișier cu numele AUTOEXEC.BAT. Acesta este un fișier text în care utilizatorul trece toate comenzile DOS care se doresc a fi executate la lansarea sistemului. De obicei, aceste comenzi se referă la:

- fixarea de către utilizator a datei și orei exacte de pornire;
- fixarea modului de afișare a **prompterului** de invitare la comenzi;
- precizarea directoarelor de pe disc unde DOS caută comenzile externe.

La fel ca și CONFIG.SYS, fișierul AUTOEXEC.BAT poate fi pregătit (eventual în avans) folosind un editor de texte.

- e) **Copierea fișierelor dorite pe discul sistem.** Se realizează cu comanda XCOPY.

Sistemul DOS dispune de comanda internă SELECT destinată realizării automate a ultimelor 3 etape.

### **Organizarea și specificarea fișierelor DOS**

Sistemul de operare DOS, începând cu versiunea 2.0 adoptă un sistem de fișiere organizat arborescent. În forma cea mai simplă, un fișier obișnuit se specifică astfel: **nume. tip**, unde **nume** are maximum 8 caractere, iar **tip** are maximum 3 caractere.

Fișierele speciale DOS au nume predefinite, care nu pot fi utilizate ca nume pentru fișierele obișnuite. Numele rezervate sunt:

- CON desemnează tastatura, dacă este folosit ca fișier de intrare, respectiv ecranul terminalului, dacă este folosit ca fișier de ieșire;
- PRN (având ca sinonim LPT1), LPT2, LPT3 desemnează una dintre imprimantele paralele;
- AUX (având ca sinonim COM1), COM2, COM3 și COM4 desemnează unul dintre adaptoarele de comunicații asincrone);

- NUL este numele unui fișier (echipament) fictiv. Folosit ca fișier de intrare generează imediat pentru sistemul de gestiune a fișierelor marcatorul de sfârșit de fișier, iar folosit ca fișier de ieșire, informațiile “scrise” în el nu sunt de fapt depuse nicăieri.

În forma cea mai generală, un fișier pe suport disc se specifică prin cele patru zone, adică: [**periferic**] [**cale**] **nume** [**.tip**], unde:

- **periferic** se va specifica printr-o literă urmată obligatoriu de “:”. DOS poate avea maximum 26 de periferice de tip disc notate cu literele alfabetului, începând cu A. În mod obligatoriu literele A: și B: desemnează dischete. Dacă sistemul de calcul dispune de hard disc el (primul dintre ele) va fi notat cu C:, apoi pot urma și alte periferice de tip disc. Dacă lipsește specificarea pentru periferic, sistemul presupune că este vorba despre perifericul implicit;
- **cale** indică, după caz, succesiunea directorilor de la rădăcina până la fișier (primul simbol din **cale** este “\”) sau de la directorul curent până la fișier (când **cale** nu începe cu “\”). Separarea directorilor DOS se face cu simbolul “\”. Dacă specificarea de **cale** lipsește, atunci sistemul consideră că fișierul face parte din **directorul curent**. Sistemul DOS reține câte un director curent pentru fiecare periferic disc din sistem și de aceea, atunci când se trece de la un periferic la altul fără să i se indice acestuia **calea**, sistemul consideră că este vorba despre directorul curent al perifericului respectiv.
- **tip** poate lipsi, dar în general se folosește pentru a indica conținutul fișierului. Câteva tipuri sunt uzuale: BAS, FOR, PAS, C, ASM, OBJ, EXE, COM, BAT, LST, DAT, DOC, TXT, MAP, LIB, TMP, BAK.

Unele comenzi DOS acceptă specificarea generică a unei familii de fișiere. Aceasta presupune folosirea, numai în zona de **nume** sau **tip** (eventual în ambele) a simbolurilor speciale: “\*”, care poate substitui un grup de caractere, sau “?”, care poate substitui un caracter pe poziția pe care se găsește.

Iată câteva exemple de nume de fișiere:

PROG.PAS, PROG;

B: ALFA. UNU, B: ALBA. DOI, B: ALGA. COM;

C: † programe† pascal† \*. PAS

Primele două fișiere se află pe perifericul implicit și în directorul curent. Următoarele trei fișiere sunt pe discul B: și în directorul implicit. Specificarea generică PROG.\* include primele două fișiere. Specificarea B: AL?A.\* conține cele trei fișiere de pe discul B:.

Specificarea C: \ programe\ pascal\ \*. PAS conține toate textele sursă Pascal aflate în directorul numit **pascal**, care este subdirector al directorului

**programe**, care este subdirector al rădăcinii discului C:. Dacă directorul curent este \ **programe**, atunci specificarea poate fi făcută și sub forma C: \ pascal\ \*. PAS.

### **Structura internă a unui disc DOS**

Un sector al unui disc este fie liber, fie ocupat de un fișier anume, fie conține informații destinate special sistemului de operare.

În sistemul DOS, unitatea de alocare a spațiului pe disc este **cluster-ul**. Un cluster este o succesiune de 1 până la 8 sectoare vecine, alocate împreună aceluiași fișier. Dimensiunea unui cluster este o constantă care depinde de tipul discului și de tipul sistemului de calcul. Valoarea acestei constante este în cazurile următoare:

- 1 sector / cluster, pentru o dischetă simplă față;
- 2 sectoare / cluster, pentru o dischetă dublă față;
- 4 sectoare / cluster, pentru un hard disc al unui IBM PC / AT;
- 8 sectoare / cluster, pentru un hard disc al unui IBM PC / XT;

Pentru a vedea modul în care un anumit fișier utilizează clusterii unui disc fizic, este necesar să parcurgem mai multe etape, care în abordarea top-down (de sus în jos) ar fi:

- a) partiționarea unui hard disc în discuri logice;
- b) structura și alocarea spațiului la un disc logic;
- c) structura și reprezentarea unui director;
- d) manipularea fișierelor și articolelor.

### **Structura și alocarea spațiului la un disc logic**

Un disc logic este fie o dischetă fizică, fie o partiție de hard disc, fie un ramdrive (disc simulat în memoria RAM). Un disc logic este împărțit în patru zone și anume:

- a) **Zona de boot** conține informații care sunt utilizate când se încarcă sistemul de operare. Aceste informații sunt dependente de sistem, dar în mod tipic conțin un program de încărcare;
- b) **FAT - Tabela de alocare a fișierelor** conține informații cu ajutorul cărora se gestionează spațiul pe disc. În FAT este înregistrată starea zonelor de pe disc divizate în entități numite cluster. Pentru a cunoaște lungimea unui cluster indiferent de disc se va specifica comanda CHKDSK. Alocarea clusterelor se realizează prin intermediul FAT care conține o listă de numere care reprezintă fiecare cluster alocat pe disc:
  - dacă intrarea în FAT este 0, atunci clusterul este liber;

- dacă intrarea în FAT este nenulă, atunci este un cluster în folosință, iar numărul găsit este utilizat pentru a lega diferite clustere care alcătuiesc împreună fișierul;
- c) **ROOT directory - directorul rădăcină**, simbolizat prin “**⚭**” este un director în care se memorează, pentru fiecare fișier următoarele informații:
- numele și extensia sa;
  - data creării sau a modificării (vizibile cu comanda DIR);
  - numărul de început al clusterului (clusterul care conține prima înregistrare a fișierului);
  - atributul fișierului;
- d) **File - Zona alocată fișierelor** conține în prima parte fișierele sistemului de operare IBMBIO. COM și IBMDOS. COM și în continuare, fișierele divizate în clustere a căror ordine de legături este păstrată în FAT. Subdirectorii sunt localizați în FAT ca orice alt fișier, iar atunci când se utilizează conținutul unui subdirector, acesta acționează ca și directorul rădăcină.

## Principalele comenzi DOS

Interpretorul de comenzi COMMAND. COM se încarcă în memorie la pornirea sistemului de operare. El ține evidența discurilor implicite și a directoarelor curente de pe fiecare disc și în plus, efectuează un număr de comenzi DOS prestabilite.

Comenzile DOS se împart în două categorii:

- a) **comenzi rezidente (interne)**, sunt comenzile înglobate în COMMAND. COM;
- b) **comenzi tranzitorii (externe)**, sunt cele care solicită existența pe disc a unui fișier de tip .COM sau .EXE având ca **nume** numele comenzii.

O comandă rezidentă se lansează astfel: ... > **comanda argumente**, iar o comandă rezidentă: ... > **comanda [d:] [cale] argumente**.

Ca efect, COMMAND.COM caută și reperează pe disc un fișier având unul din numele: **comanda.COM, comanda.EXE, comanda.BAT**

Dacă se specifică “**d:**”, atunci căutarea se face pe discul specificat, în caz contrar se caută pe discul implicit.

Dacă se specifică “**cale**”, atunci fișierul se caută în directorul indicat de **cale**.

În situația în care lipsește atât specificarea de disc cât și specificarea de **cale**, căutarea se face în ordine:

- în directorul curent;

- în directoarele specificate prin comanda PATH.

În cazul în care, în același director este reperat mai mult de un fișier din cele trei, de exemplu, unul de tip COM și altul de tip BAT, prioritatea la lansare o are COM, apoi EXE și la urmă BAT.

O sinteză a formatului comenzilor DOS cu principalele opțiuni este prezentată în figura următoare. Comenzile interne sunt evidențiate prin litera “I” plasată în dreapta comenzii.

```

CD [d:][cale] sau CHDIR [d:][cale] I
MD [d:][cale] sau MKDIR [d:][cale] I
RD [d:][cale] sau RMDIR [d:][cale] I
JOIN d1 d2 \director [/D]
FDISK [d:]
FORMAT d: [/S][V]
UNFORMAT d:
LABEL [d:][etichetă_volum]
DISKCOMP d1: d2:
DISKCOPY d1: d2:
CHKDSK [d:][cale][fișier[. tip]][/F][V]
RECOVER d:
DIR [d:][cale][fișier[. tip]][/P][W] I
COPY [/A][B][d:][cale]fișier[. tip]
    [+d:][cale]fișier[.tip]...[d:][cale]fișier[.tip] I
XCOPY [d:][cale]fișier[. tip] [d:][cale]fișier[. tip]
    [/A][D: ll-zz-aa][E][M][P][S][V][W]
REN sau RENAME [d:][cale]fișier[.tip] [d:][cale]fișier[.tip] I
DEL sau ERASE [d:][cale]fișier[.tip] I
UNDELETE [d:][cale]fișier[.tip][LIST][ALL]
COMP [d:][cale]fișier[.tip] [d:][cale]fișier[.tip]
TYPE [d:][cale]fișier[.tip] I
PRINT [d:][cale]fișier[.tip] ... [/C][P][T]
SORT [/R][+n]
FIND [V][C][N] “șir” sau [d:][cale]fișier[.tip] ...
MORE
EXE2BIN [d:][cale]fișier. COM [d:][cale]fișier. EXE
PATH [[d:]cale[;[d:]cale ...]] I
APPEND [[d:]cale[;[d:]cale ...]]
FASTOPEN d: [= n]
DATE [ll-zz-aa] I
TIME [hh-mm[:ss]] I
BREAK [ON|OFF] I

```



```

PROMPT [text]          I
SET [nume = [valoare]] I
CLS
MEM [/P]/[C]/[D]
VER                    I
VOL [d:]              I
HELP [comanda]

```

Principalele comenzi DOS pot fi grupate astfel:

a) **Comenzi de manipulare a directorilor:**

- CD (CHDIR): schimbă directorul curent în cel specificat;
- MD (MKDIR): crează un director nou, numai dacă nu există deja un director cu numele specificat;
- RD (RMDIR): șterge un director vid (nu conține nici-un fișier sau subdirector specificat);
- JOIN: realizează conectarea unei structuri de fișiere aflată pe un disc “d1” la structura de fișiere a discului “d2”. În urma comenzii, cele două sisteme de directoare sunt privite ca și unul singur, directorul rădăcină al discului “d1” fiind conectat ca subdirector cu numele “director” al rădăcinii lui “d2”. Opțiunea “/D” realizează operația inversă, adică deconectarea, iar atunci se vor specifica numai “d1” și “/D”;

b) **Comenzi de manipulare a unui disc:**

- FDISK: realizează partajarea unui hard disc în partiții;
- FORMAT: inițializează discul specificat prin “d:” la un format standard acceptat de către DOS. După ce se scrie pe tot discul, se crează directorul rădăcină și se inițializează directorul rădăcină precum și tabela FAT. Opțiunea “/S” cere ca discul respectiv să fie făcut disc sistem, opțiunea “/V” specifică o etichetă de volum pentru identificare;
- UNFORMAT: permite refacerea unui disc formatat, indicat prin “d:”. Comanda este valabilă la DOS 5.0, iar refacerea se realizează prin formatarea doar a părții de început, până la directorul rădăcină inclusiv. Înainte de scriere, această porțiune se salvează într-o zonă rezervată a discului. Utilizatorul trebuie să lanseze comanda UNFORMAT înainte de a scrie ceva pe discul formatat, iar dacă nu se respectă această condiție, sau se lansează FORMAT cu “/U”, atunci refacerea nu mai este posibilă;

- LABEL: permite crearea sau ștergerea unei etichete de volum. Această etichetă poate fi dată și prin comanda FORMAT, printr-un dialog între sistem și utilizator;
- DISKCOMP: permite compararea dischetelor indicate prin “d1” și “d2”. Comparația se face pistă cu pistă, semnalându-se diferențele. De regulă, această comandă se dă pentru a se verifica dacă s-a făcut corect o operație de copiere cu comanda DISKCOPY;
- DISKCOPY: copiază la nivel fizic, discheta “d1” pe discheta “d2” (dacă e nevoie, discheta “d2” este formatată automat);
- CHKDSK: analizează structura informației de pe un disc DOS și afișează un raport asupra stării discului. Opțiunea “/F” cere corectarea (pe cât posibil a) structurii încât să devină una validă, iar opțiunea “/V” afișează o serie de detalii în timpul execuției comenzii. Dacă se specifică un fișier (sau o familie), atunci se indică numărul de zone necontigue ocupate de fișierul (fișierele) în cauză.
- RECOVER: recuperează fișierele de pe disc care conțin sectoare defecte, și de asemenea, se marchează sectoarele defecte. Fișierele recuperate primesc nume noi, sub forma: FILEnnnn. REC, unde “nnnn” este un număr de ordine. Toate fișierele recuperate vor fi plasate în directorul rădăcină.

c) **Comenzi referitoare la fișiere:**

- DIR: listează toate intrările din fișierul director sau numai cele specificate. Parametrul “/P” generează execuția unei pauze ori de câte ori ecranul este plin de informații, iar parametrul “/W” produce o afișare condensată a directorului, listând doar numele fișierelor, câte cinci nume pe linie;
- COPY: copiază unul sau mai multe fișiere pe un alt disc și opțional, atribuie un nume diferit pentru copie. De asemenea, COPY poate crea o copie pe același disc. În timpul procesului de copiere se poate realiza și o concatenare de fișiere, dacă se specifică operatorul “+”. Parametrii “/A” și “/B” indică volumul de date care se va copia și anume:
  - în intrare: “/A” exclude copierea marcatorului de sfârșit de fișier, iar “/B” impune copierea întregului fișier;
  - în ieșire: “/A” adaugă marcatorul de sfârșit de fișier, iar “/B” nu adaugă marcatorul de sfârșit de fișier;
- XCOPY: permite copierea selectivă de fișiere sau grupuri de fișiere care pot să includă și subdirectoare. Ca opțiuni, pot fi:
  - “/A”: copierea fișierelor marcate pentru arhivare;

- “/M”: identic cu “/A”, dar schimbă marcajul de arhivare;
- “/D”: copierea fișierelor modificate după data specificată;
- “/E”: crează la destinație subdirectoarele din sursă;
- “/P”: copierea cu confirmare pentru fiecare fișier;
- “/S”: copierea din sursă a directorilor și subdirectoarelor
- “/V”: verificare scriere la destinație (recitare și comparare)
- REN (RENAME): schimbă numele fișierului specificat;
- DEL (ERASE): șterge fișierul cu numele specificat, din directorul și unitatea specificate sau implicite;
- UNDELETE: permite refacerea fișierelor șterse (valabilă la DOS 5.0). În mod obișnuit se reface un singur fișier, dar opțiunea “/ALL” permite refacerea tuturor fișierelor posibile, caz în care prima literă a numelor acestor fișiere este stabilită automat. Opțiunea “/LIST” permite afișarea fișierelor care pot fi refăcute;
- COMP: compară două fișiere de aceeași lungime, prin citirea și compararea conținutului lor. Compararea se termină fie la sfârșit, fie la depistarea a maximum 10 octeți diferiți ( primele 10 diferențe sunt afișate);
- TYPE: afișează conținutul fișierului specificat pe dispozitivul standard de ieșire;
- PRINT: tipărește o coadă (listă) de fișiere de date la imprimantă în timp ce pe sistem se pot executa alte lucrări;

d) **Filtre DOS:**

- SORT: citește date de la echipamentul standard de intrare, sortează datele, apoi scrie datele sortate la echipamentul standard de ieșire. Sortarea se efectuează crescător după codurile ASCII ale caracterelor, sau descrescător (“/R”);
- FIND: trimite la echipamentul standard de ieșire toate liniile din fișierul specificat în comandă și care conțin un șir de caractere specificat. Parametrul “/V” determină afișarea tuturor liniilor care nu conțin șirul specificat, iar parametrul “/C” permite afișarea unui contor cu numărul de apariții a șirului specificat;
- MORE: este un filtru care citește date de la intrarea standard și le transpune pe ecran, iar când acesta este plin, se așteaptă apăsarea unei taste pentru a se trece la ecranul următor;
- EXE2BIN: este un filtru special care transformă un program executabil de tip EXE în unul de tip COM;

e) **Comenzi de configurare dinamică DOS:**

- PATH: fixează căile de căutare a fișierelor executabile;
- APPEND: fixează căile de căutare a fișierelor neexecutabile;
- FASTOPEN: optimizează accesul la fișierele și directoarele folosite mai des. În acest scop se gestionează o zonă tampon cu adresele disc recent utilizate, potrivit principiului memoriei **cache**. Parametrul “n” indică numărul de adrese gestionate, cuprins între 10 și 999, cu valoarea implicită 34;
- DATE, TIME, BREAK, PROMPT: se referă la schimbarea datei, a orei exacte, a sesizării sau nu a lui <CTRL/Break>, sau a formei prompterului de invitare la comenzi;
- CLS: provoacă ștergerea completă a ecranului terminalului
- SET: fixează valori ale unor parametri aflați într-o zonă de memorie controlată de COMMAND. COM.

f) **Comenzi de informare:**

- MEM: afișează conținutul memoriei la un moment dat;
- VER: afișează versiunea sistemului de operare DOS;
- VOL: afișează eticheta de volum specificat sau a celui curent;
- HELP: oferă detalii despre comenzile DOS în general, sau despre o anumită comandă DOS specificată.

**Fișiere de comenzi (BAT) sub DOS**

Un fișier de comenzi (al cărui tip este .BAT), este un fișier text care conține una sau mai multe comenzi și/sau directive care se execută succesiv, fără intervenția directă a utilizatorului. Un prim exemplu de fișier de comenzi DOS este fișierul AUTOEXEC. BAT. Dacă acest fișier este prezent în directorul rădăcină pe discul de pe care se face încărcarea, DOS execută automat comenzile din el la fiecare încărcare a sistemului.

Un fișier de comenzi DOS conține mai multe tipuri de construcții:

- comenzi DOS propriu-zise;
- etichete;
- caracterele speciale |, <, > și %;
- parametrii formali care vor fi înlocuiți la lansare cei actuali;
- variabile globale rezidente în interpretorul de comenzi;
- variabile locale utilizate în instrucțiuni repetitive;
- directive adresate interpretorului fișierelor de comenzi.

*Exemplu:*            TIPAR. DAT:    DIR \*. PAS

COPY \*.PAS a:

DEL \*.PAS

O **etichetă** se scrie sub forma: **: nume**, unde caracterul “:” este urmat de maximum 8 litere sau cifre. O etichetă apare pe o singură linie, la început de rând.

**Caracterele speciale** |, < și > sunt interpretate în comenzi prin legare în pipe a două comenzi, respectiv redirectarea intrării și/sau ieșirii standard a unei comenzi.

**Caracterul %** marchează parametri formali, variabilele globale și variabilele locale.

Într-un fișier de comenzi pot fi specificați până la 10 parametri formali, notați %0, %1, ..., &9. Corespunzător acestora, în linia de comandă care lansează în execuție fișierul de comenzi argumentele sunt puse în corespondență cu acești parametri.

### **Directivele interpretorului fișierelor de comenzi**

- FOR %% variabila IN (mulțime) DO comanda;
- CALL fișier\_comenzi;
- IF [NOT] ERRORLEVEL n comanda, sau
- IF [NOT] șir1 == șir2 comanda sau
- IF [NOT] EXIST fișier
- GOTO etichetă
- SHIFT
- ECHO [ON|OFF|mesaj]
- PAUSE [mesaj]

Directiva **FOR** permite prelucrarea iterativă a unei comenzi dintr-un fișier de comenzi. %%**variabilă** este o variabilă locală de ciclare care va fi înlocuită secvențial cu fiecare membru din **mulțime**, apoi comanda este evaluată și executată. Dacă un membru al mulțimii este o expresie construită generic (cu \* și/sau ?), atunci %%**variabilă** va fi înlocuită pe rând cu toate numele de fișiere care se potrivesc cu această expresie.

De exemplu, comanda:

```
FOR %%f IN (*.PAS *.DOC) DO TYPE %%f
```

va tipări pe ecranul terminalului mai întâi toate fișierele de tip PAS, apoi pe cele de tip DOC.

**Directiva CALL** permite apelul (lansarea în execuție) a unui fișier de comenzi din altul. La terminarea fișierului apelat, controlul revine fișierului apelant. Este posibil ca ultima linie a unui fișier de comenzi să ceară lansarea altui fișier de comenzi (fără CALL), și în acest caz se realizează o înlănțuire fără revenire a celor două fișiere.

**Directiva GOTO** transferă controlul în fișierul de comenzi, în linia de comandă situată după eticheta specificată.

**Directiva IF** permite execuția condiționată a unei comenzi DOS din fișierul de comenzi, sau a unei alte directive.

Fiecare comandă DOS transmite, în momentul terminării, un cod cu o valoare între 0 și 255. De obicei, un cod cu o valoare nenulă indică o situație neobișnuită, în anumite situații de programare specificând ramura pe care s-a terminat programul. Condiția **“ERRORLEVEL număr”** este adevărată dacă codul de retur al ultimei comenzi este mai mare sau egal cu numărul scris după cuvântul ERRORLEVEL.

Condiția **“șir1 == șir2”** este adevărată dacă valorile celor două șiruri de caractere coincid, iar cel puțin unul din șiruri trebuie să conțină un parametru formal sau o variabilă globală.

Condiția **“EXIST fișier”** este adevărată dacă fișierul specificat, simplu sau generic există.

Acțiunea directivei **IF** fără **NOT** înseamnă execuția comenzii dacă condiția este adevărată, iar dacă **NOT** este prezent, atunci comanda se execută când condiția este falsă.

**Comanda ASK** face parte dintr-o familie de programe utilitare oferite de firma NORTON. Prin intermediul acestei comenzi, utilizatorul poate interacționa cu fișierul de comenzi.

**ASK mesaj, [litere]** are ca efect afișarea pe ecran a **“mesaj”**, care nu conține **“,”**, după care așteaptă ca utilizatorul să apese una din literele dintre paranteze . Dacă utilizatorul a tastat a i-a literă din listă, atunci **ASK** se termină cu **ERRORLEVEL = i**. Evident, o succesiune de directive **IF** poate realiza o ramificare multiplă a execuției.

Exemplu: ASK.BAT:

```
ASK   Doriți testare AV?, [DdNn]
IF ERRORLEVEL 3 GOTO :F{R{
      SCAN   c: d:
      PAUSE
      : F{R{
```

**Directiva SHIFT** permite folosirea a mai mult de 10 parametri formali. Efectul ei este acela de a **“decala”** parametrii formali, făcând să se piardă parametrul %0, parametrul care înainte era %1 să devină %0, cel ce fusese %2 să devină %1 etc.

Exemplu: STERGE.BAT:

```
:IAR
SHIFT
IF “%0” == “ “ GOTO :STOP
```

```
DEL %0
GOTO :IAR
:STOP
```

Lansat cu ...> STERGE a b c d e f g h i j k l va șterge pe rând cele 12 fișiere citate în comandă.

**Directiva ECHO** permite să se afișeze (ON) sau să nu se afișeze (OFF) comenzile care se execută, sau dacă directiva conține un mesaj, se afișează acest mesaj pe ecran.

**Directiva PAUSE** suspendă execuția fișierului de comenzi până la apăsarea unei taste.

Întreruperea execuției unui fișier de comenzi se face cu <CTRL/C>.

### Sistemul Windows

**Windows** poate fi considerat ca o extensie a sistemului de operare DOS, menit pe de o parte să pună în valoare calitățile DOS încă neexploatate, iar pe de altă parte să asigure utilizatorului o interfață grafică deosebit de atractivă. Această interfață este asigurată atât față de DOS, cât și față de programele de aplicații, indiferent dacă acestea sunt produse de firmă sau sunt dezvoltate de către utilizator.

Trebuie precizat că **Windows** nu este numai o **suprafață de operare**, ci o extensie efectivă a DOS. În susținerea acestui fapt trebuie arătat că **Windows** oferă trei facilități care direct sub DOS se pot realiza numai cu mare efort din partea utilizatorului:

- a) execuția (aproximativ) “simultană” a mai multor aplicații;
- b) transmiterea sub controlul utilizatorului a unor date între aplicații;
- c) realizarea unui mecanism standard de interfață grafică utilizator, la toate nivelele de aplicare.

Sub **Windows** pot fi lansate în lucru mai multe aplicații. O aplicație lansată poate fi **suspendată temporar**. După suspendare poate fi lansată o altă aplicație, sau poate fi relansată una dintre cele suspendate. Este deci limpede că nu este vorba de **multitasking** în sensul clasic. Dacă un astfel de multitasking este necesar sau nu, aceasta depinde de aplicații. În orice caz, deși în documentații se vorbește de **multitasking** sub **Windows**, acesta trebuie înțeles în sensul de mai sus.

Singurele aplicații efectiv executabile în paralel sunt cele oferite de DOS:

- a) listarea la imprimantă în paralel cu execuția altor programe;
- b) supravegherea ceasului sistem în vederea lansării unui program la o oră fixată.

## Realizarea schimbului de date

**Windows** gestionează implicit o aplicație numită **Clipboard**. Prin intermediul acesteia utilizatorul poate depozita temporar, din orice aplicație, informații din zona rezervată **Clipboard**. Informația poate fi de tip text sau grafică. Tot prin **Clipboard** utilizatorul poate edita conținutul informației depozitate. În sfârșit, din orice aplicație se pot extrage informații depozitate în zona **Clipboard**. În acest mod se pot transmite informații de la o aplicație la alta.

## Interfața grafică Windows (GUI - Graphics User Interface)

Elementul esențial de intrare în această interfață este **tastatura**. Pentru simplificarea muncii, **Windows** “preferă” pentru cele mai multe răspunsuri simple unul sau două apăsări ale unui buton de la mouse.

Elementul esențial de afișare în această interfață este **fereastra**. Aceasta este o porțiune dreptunghiulară afișată într-o zonă a ecranului, destinată comunicării utilizator-**Windows**. Principalele componente ale unei ferestre sunt:

- a) **Titlul** ferestrei;
- b) **Butonul de închidere / deschidere** a ferestrei (stânga sus);
- c) **Butonul de minimizare** (triunghiul cu vârful în jos plasat în partea dreaptă sus);
- d) linia pentru **meniuri pull-down** (File, Window, Help etc.);
- e) **scroll-bar** pentru defilare verticală (plasat pe latura verticală din dreapta și prevăzut cu două săgeți pentru ridicare sau coborâre și un buton nescris pentru a marca poziția curentă);
- f) **scroll-bar** pentru defilare orizontală (plasat pe latura orizontală de jos, cu același conținut ca și **scroll-bar** pentru defilare verticală);
- g) **conținutul propriu-zis** al ferestrei care este de fapt partea cea mai mare a ei, plasat între cele două **scroll-bar** și linia de meniuri **pull-down**.

La un moment dat pe ecran pot să existe mai multe ferestre, eventual suprapuse total sau parțial. Fiecare fereastră aparține unei aplicații. Fereastra care este deasupra și în care apare cursorul este cea activă, odată cu aplicația căreia îi aparține. Activarea/dezactivarea unei ferestre se face printr-o simplă apăsare pe butonul din stânga sus de la mouse.

Cu ajutorul mouse-ului fereastra activă poate fi deplasată sau redimensionată. Pentru aceasta se deplasează mouse-ul la una din margini, se lasă până când cursorul își schimbă forma, după care, cu butonul ținut apăsat se deplasează și se dimensionează după dorință.

**Scroll-bar**-urile apar și se folosesc atunci când spațiul oferit de partea principală a ferestrei este prea mic pentru a putea afișa toată informația. Din această cauză, se afișează doar o parte din informații, numai cât încap în



fereastră. Apăsând butonul mouse-ului cu cursorul pe una dintre aceste bare de defilare, informația se va deplasa făcând vizibile alte informații în fereastră.

O fereastră poate fi **închisă** (close), caz în care aplicația proprietar se termină. **Suspendarea** se realizează fie suprapunând peste ea o altă fereastră, fie minimizând fereastra respectivă.

O fereastră **minimizată** apare pe ecran sub forma unui așa-zis **icon** (**pictogramă**). Aceasta este o imagine de dimensiuni mici care conține un mic desen sugerând aplicația.

**Maximizarea** unei ferestre plecând de la o pictogramă înseamnă “trezirea” aplicației respective, maximizarea înseamnă dimensionarea ei astfel încât să ocupe tot ecranul.

Un meniu **pull-down** devine activ dacă se apasă butonul din stânga de la mouse atunci când cursorul se află pe numele meniului. În acest moment apare o fereastră care permite utilizatorului să aleagă o acțiune din mai multe posibile. Utilizatorul poate tasta pentru alegere litera subliniată, sau se poate deplasa cu săgețile sau cu mouse-ul pe titlul variantei alese, după care poate tasta <ENTER> sau un “clic” (apăsarea unuia dintre butoanele de la mouse).

## Resurse Windows

Conținutul principal al ferestrei depinde de tipul acesteia. Fiecare tip de fereastră constituie pentru Windows o **resursă**. O resursă **Windows** este o structură de date care definește o porțiune vizibilă pe ecran a unui program. O astfel de resursă este de regulă materializată printr-un fișier de un anumit tip.

Esența programării sub **Windows** constă în următoarele: un program simplu DOS de tip EXE, înainte de a putea fi executat sub **Windows** este supus unei faze suplimentare numită **adăugarea resurselor la fișierul executabil**.

Fiecare dintre mediile de programare celebre oferă biblioteci de interfață **Windows**. Aproape toate acestea adoptă pentru programare politica de programare orientată obiect OOP (Object Oriented Programming). Spre exemplu, firma **Borland** a lansat o dată cu **Turbo Pascal 6.0** pachetul **Turbo Vision**. Aceasta permite o interfață excelentă la nivel text cu ecranul. Curând după aceasta, aceeași firmă lansează **Object Vision**, care oferă o serie de unități de lucru de interfață **Windows** în aceleași scopuri. Pachete similare au fost lansate pe piață pentru C, C++, **dBASE**, **Paradox** etc.

Utilizatorii nu întâmpină greutăți deosebite pentru programare sub Windows. De exemplu, un program Pascal obișnuit se scrie la fel mediul **Turbo** obișnuit ca și în mediul **Turbo for Windows**, singura modificare este declararea **uses** de folosire a unității **Wincrt**, deci programul va avea forma:

**program**            nume;  
**uses**                **Winert**  
...  
**end.**

Trebuie totuși precizat un lucru în legătură cu programarea sub **Windows**. Nucleul acesteia pretinde un control total asupra resurselor hard ale sistemului de calcul. Micile “șmecherii” posibile sub DOS: interceptarea tastaturii, definierea unor întreruperi ale utilizatorului, deturnarea unor întreruperi care vizează ecranul și tastatura etc. nu mai sunt posibile.

Principalele tipuri de resurse **Windows** sunt:

- a) acceleratori;
- b) bitmapuri (hărți de biți);
- c) cursoare;
- d) ferestre de dialog (dialog boxes);
- e) fonturi;
- f) iconuri (pictograme);
- g) meniuri;
- h) tabele de stringuri;
- i) resurse definite de utilizator;

Acțiunile posibile cu resursele sunt:

- a) adăugarea unei resurse la un proiect;
- b) ștergerea unei resurse;
- c) redenumirea unei resurse;
- d) editarea unei resurse;
- e) salvarea unei resurse;
- f) specificarea opțiunilor de memorie ale unei resurse.

Să vedem acum ce este de fapt fiecare tip de resursă **Windows**. Înainte de a le defini, trebuie spus că Windows oferă instrumente grafice de definire a lor pe bază de dialog. Utilizatorul poate de asemenea să le descrie prin intermediul unui limbaj specializat. Experiența a demonstrat că în marea majoritate a cazurilor, resursele sunt definite nu prin descrieri anticipate în limbajul specializat ci prin intermediul instrumentelor de dialog oferite de **Windows**.

**Accelerator** este o combinație de chei pe care utilizatorul o definește pentru a comanda efectuarea unei sarcini dintr-o aplicație.

De exemplu, dacă într-o aplicație are nevoie de ștergerea frecventă a mai multor fișiere, el poate să-și definească combinația <ALT/F8> pentru a șterge fișierul punctat de cursor (este vorba aici de prescurtarea unor comenzi, nu de mărirea vitezei de execuție).

**Bitmap** are rolul de a afișa imagini grafice în programul **Windows**. Crearea unui bitmap se face cu ajutorul componentei **Paint Editor** care

crează în acest sens un fișier. Bitmap-urile sunt folosite pentru a reprezenta: săgețile scroll-bar, butoanele de minimizare și maximizare și orice alte imagini create de utilizator.

**Cursorul** este un bitmap alb-negru de  $32 \times 32$  biți prin care este indicată poziția curentă pe ecran. Proiectarea unui cursor se face cu **Paint Editor**.

**Dialog-Boxes** (ferestre de dialog) sunt resursele care oferă modalitatea de interacțiune cu o aplicație. O astfel de fereastră oferă pentru interfață meniuri pull-down, butoane pentru selectarea unei valori, butoane care permit selectarea simultană a mai multor valori (radio-buttons), căsuțe pentru introducerea și editarea unui text etc. Pentru creare, **Windows** oferă un editor special pentru dialoguri.

**Fonturi.** Un font este o colecție de date care descriu un caracter individual, precum și modul în care acesta va fi afișat sau tipărit.

**Pictograme.** O pictogramă [icon] este o imagine bitmap de  $32 \times 32$  biți (dacă apare și numele iconului) sau  $32 \times 16$  biți (dacă apare doar imaginea). Proiectarea unui icon se face cu **Paint Editorul**. Imaginea creată depinde de fantezia utilizatorului, totul este să sugereze, foarte sintetic, conținutul aplicației respective.

**Meniuri.** Un meniu cere utilizatorului să aleagă una dintre opțiunile oferite. Pentru crearea de meniuri, **Windows** oferă editor special.

**Tabele de stringuri.** O astfel de tabelă conține mesaje de eroare, prototipuri sau orice colecție de mesaje text pe care o aplicație le poate da. Pentru creare, există un editor special de stringuri.

**Resurse definite de utilizator.** Există posibilitatea de a defini și alte tipuri de resurse decât cele prezentate până acum. O resursă o dată definită, poate fi folosită ori de câte ori se dorește.

### **Operarea sub Windows**

Lansarea în lucru se face cu comanda: ....> **Win**. După lansare apare fereastra principală. Iconul marcat este deja prezent în memorie și poate fi făcut activ prin meniul **File**. După activare apar două ferestre parțial suprapuse. De aici se poate face activ un alt icon sau alege o altă aplicație.

Refăcând activă una dintre ferestrele “de dedesubt”, de acolo se poate alege o altă aplicație etc. Minimizând și eventual deplasând și redimensionând, se pot crea mai multe aplicații suspendate temporar și una activă.

Meniul **Window** apare în foarte multe aplicații. El permite o aranjare a ferestrelor de pe ecran fără intervenția efectivă a utilizatorului. Există în principal două moduri de aranjare:

- a) **cascadă**, parțial suprapuse, cele mai mari dedesubt, iar cele mai mici deasupra;
- b) **tile (piese de domino)**, când ferestrele sunt plasate alăturat, fără suprapuneri.