

DBMS

DBMS - structură:

1) DDL

2) DML $\begin{cases} \text{SQL} \\ \text{QBE} \\ \text{4GL} \end{cases}$

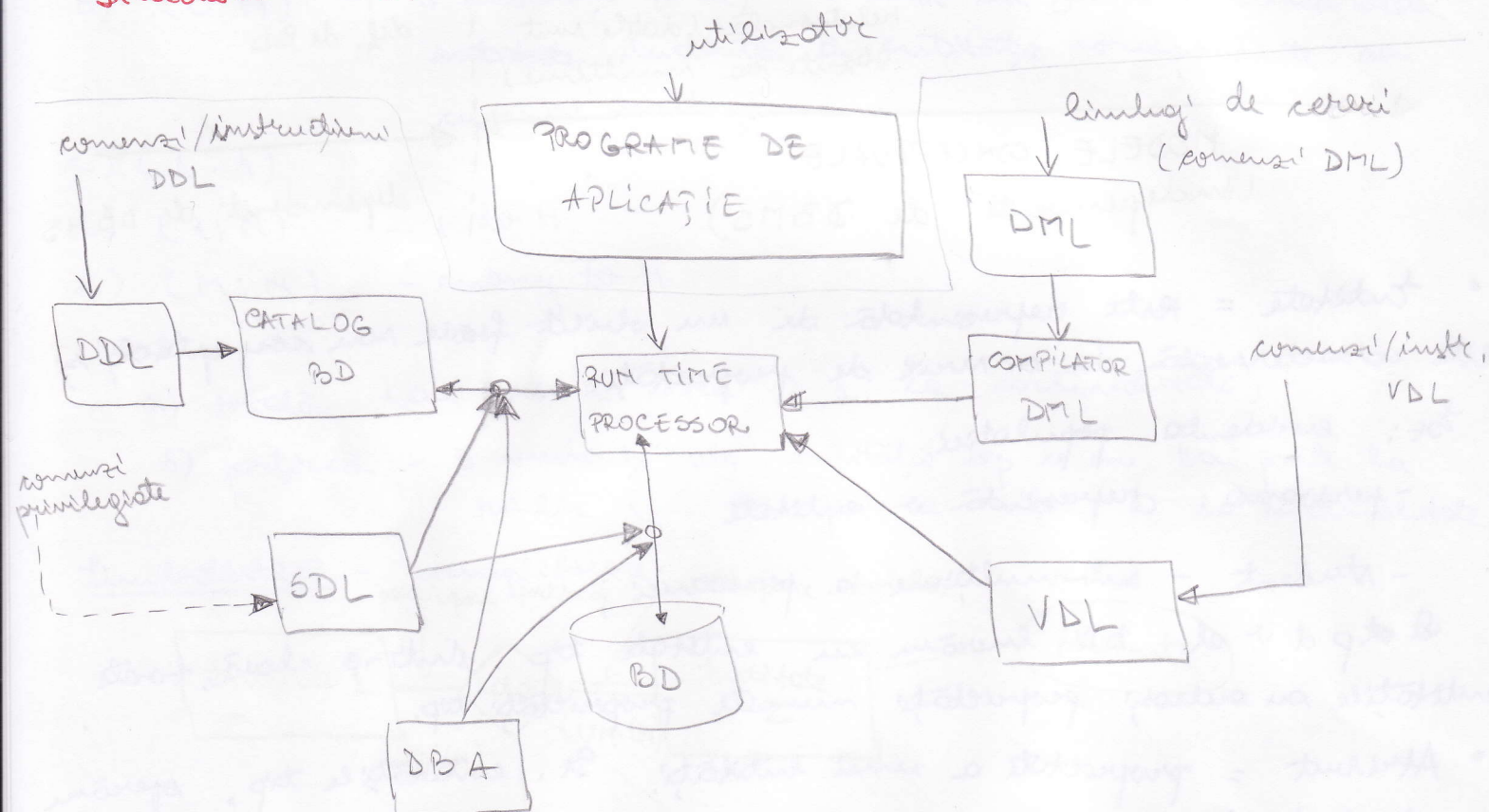
3) VDL

4) SDL - adresa B, B⁺ (metode logice de acces dependente de modul de core e organizată mem. externă)

5) DBA

1+2+3+4 → limbajul SQL

Structură DBMS:



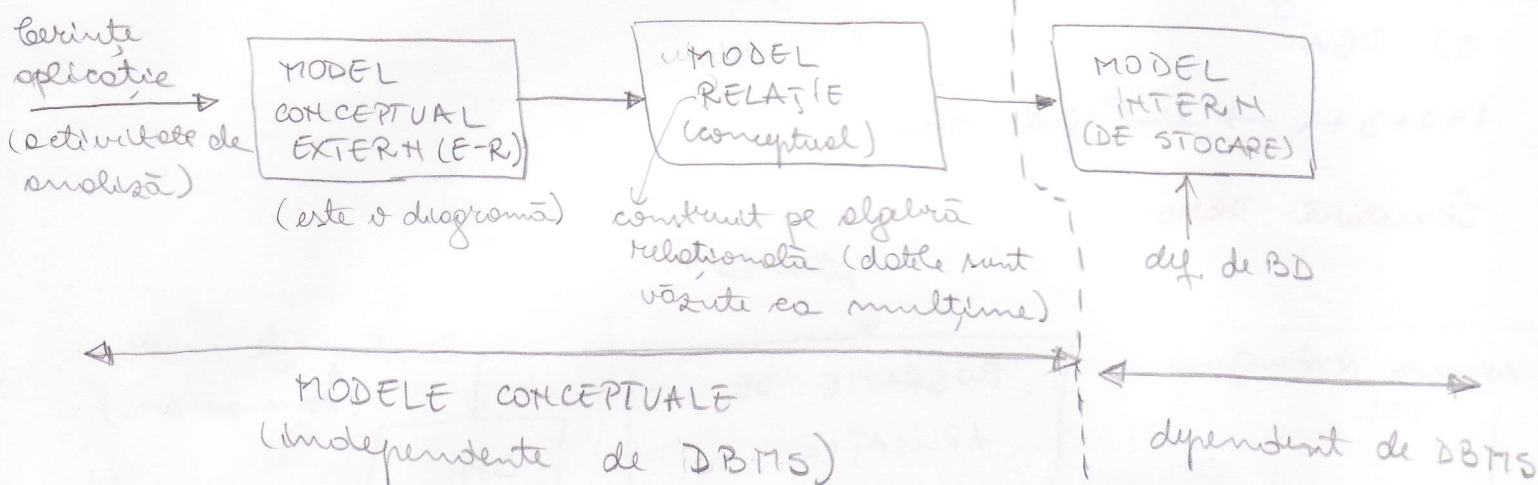
OBS: în afară de programele de aplicație, restul structurii reprezintă DBMS-ul.

3. Modelarea datelor (model E-R)

entitate relație

Este componenta din tehnologia de dezvoltare a unei aplicații de BD care pornind de la cerințe de utilizare dezvoltă o structură a BD astfel încât, prin această structură să se minimizeze redundanța, dar în același timp să asigure performanțe bune atât la secvența de prelucrare.

Minimizarea redundanței = în BD nu trebuie să avem duplicări;
 Performanțe bune = cererile să se rezolve f. rapid



• Entitate = este reprezentată de un obiect fizic sau conceptual și este caracterizată de o serie de proprietăți.

Ex: evidența populației

- persoane: reprezintă o entitate

- student - submulțime a persoanei;

Și p.d.v. al BD lucrăm cu entități tip: într-o clasă, toate entitățile au același proprietăți numite proprietăți tip.

• Atribut = proprietate a unei entități. Pt. entitățile tip, operăm cu atribute tip.

Atribute - caracterizări:

- atomic: nu poate fi descompus în componente cu semnificație

- compus: poate fi descompus în componente cu semnificație;

Un atribut compus se descompune în atribute atomice.

Exemple: atribut compus - Nume (nume-f, Inițială, Prenume)

atribut atomic Adresă (Stradă, Nr., Localitate)

cheie: un atribut este atribut cheie dacă valorile sale sunt \neq pt. fiecare instanță ale valorii se o caracterizează.

Atributul cheie poate fi < natural
artificial (CNP, cod matrică)

de regulă, pt. o entitate tip se atribuie un atribut cheie.

• Relație (asocier): definită asocierea dintre instanțele apartinând
(relationship) a 2 entități tip sau chiar mai multe.

O relație se caracterizează prin 2 parametri:

- cardinalitate: $(m:n)$, $m, n \in \mathbb{N}$

$m = nr.$ minim al instanțelor entității tip ce participă la relație

$n = nr.$ maxim al instanțelor

Știind cardinalitățile putem avea relații de tipul:

a) $(0:1)$

b) $(0:N)$ - \exists instanțe care nu sunt în relație cu nicio altă
~~instanță~~ instanță a entității corespondente sau
cu mai multe;

c) $(1:1)$

d) $(1:N)$ - 1 to N

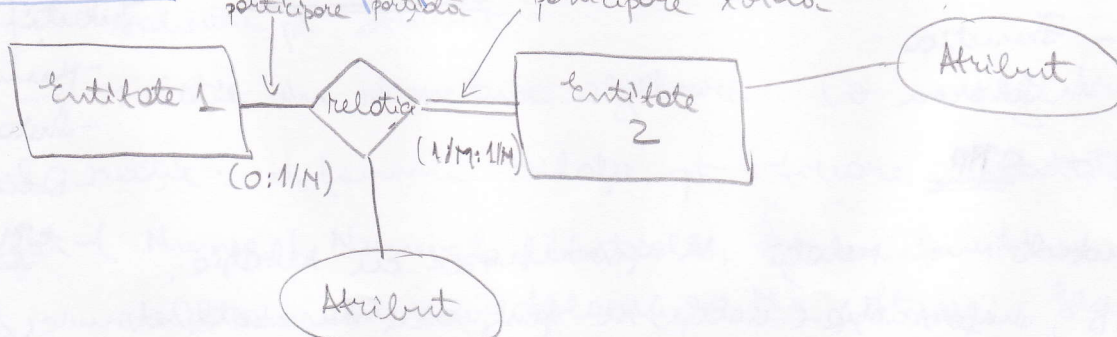
e) $(M:N)$ - many to N

- participare:

a) totală (cu primele cifre ≥ 1 la cardinalitate)

b) parțială - \exists instanțe ale entității tip ce nu sau poate la
relație (cu primele cifre 0 la cardinalitate)

Simbolistică - exemplificare:



un atribut care caracterizează relația

Se cere să se proiecteze aplicația de BD pt o companie, sau
învedere urm. elem:

- angajați sunt grupati în departamente

- un departament are maxim 4 angajați, maxim - reprezent

- atribuțiile de la cadrul companiei s.u. proiecte

- un proiect este coordonat de un departament (apartine lui)

- un departament nu să coordoneze cel puțin 1 proiect

- fiecare departament este condus de un manager

- un angajat-manager este managerul unui singur departament

- la proiecte lucrează angajați companiei

- un angajat lucrează la cel puțin ~~un~~ un proiect, dar poate
lucrea și la mai multe, indiferent de unde sunt coordonate.

Se dorește să se țină evidența nr. orelor săptămânale allocate fiecărui
proiect la care lucrează un angajat; de asemenea, să se țină evidența
persoanelor ocupate cu întreținerea angajaților → entitate sală (perso-
nele sală în baza muncii în relație cu angajații).

Apdr organizatorice, se realizează echipe sau unități în care din
angajat este supervisor, dar restul sunt supravegheați.

→ Se începe cu identificarea entităților tip:

Angajat:

- Nume
- Adresă
- Telefon
- Data nașterii
- Sexul
- Studii
- Funcția
- Salariu

(atribut cheie) - CNP

Departament:

- D-nume
- D-locatie
- Telefon
- Fax
- ad-număr

Proiect:

- P-nume
- Beneficiar
- Cost
- P-număr

Întreținut:

- Nume
- Data nașterii
- Ocupația
- CNP

OBS: - atributul unei relații se modelează cu relație.

- în BD, informațiile (date) se pot rezolva la nivel de date (dar
uneori se poate stoca și un atribut compus - depinde de funcțion)

Proiect (G nume, Beneficiar, Loc, 2 număr, Sep-Număr)

Interfăinț (CHP-i, nume, localitate, data-măști, CHP).

- prin CHP la interfăinț modelul relației între interfăinț și angajat, prin chestiunea angajului;

Relația supervisor - este necesară

It modelarea relației Ang-Proiect exemplum nouă relație:

Lucrarea (CHP, Proiect-număr, Ora).

↑
cheie pt. 'lucrarea'
↓
similar cu notele unui student.

4. Stocarea datelor, accesul la date

medii de stocare → memorie externă
BD

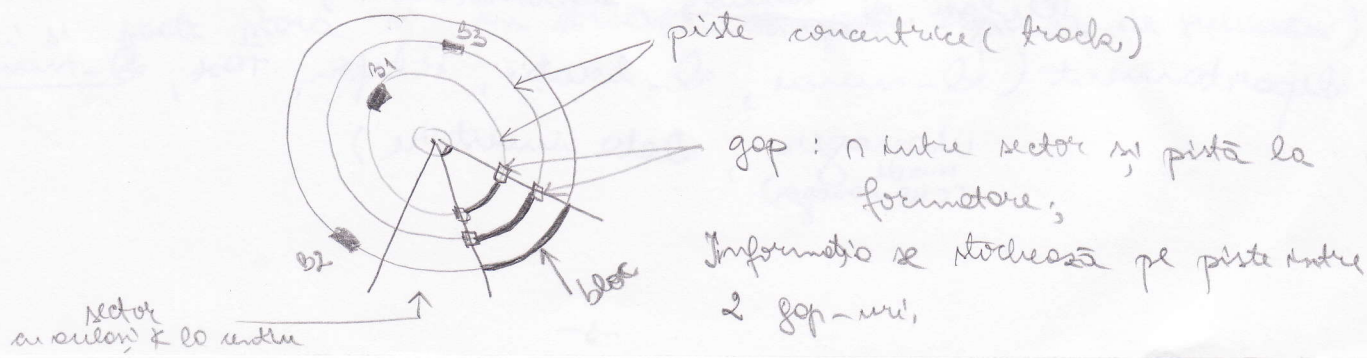
BD este: - persistentă - datele sunt stocate permanent
- volum mare al datelor

Accesul la BD reprezintă acces la mem. externă cu timp de acces mare. It. prelucrarea datelor, se utilizează memoria internă. Transferul din mem. externă la mem. internă este consumatoare de timp.

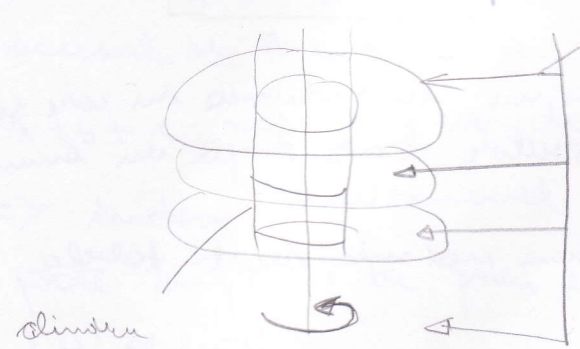
Tipuri de mem. externă

- suport magnetic < hard-disk
bandă magnetică - backup
- suport optic < CD/DVD
R/W
↳ casetă magnetică de tip R/W.

Organizare hard-disk (HD):



- + sces în mem. externă se realizează multum în bloc.
- toate blocurile au aceeași cantitate de informație; densit. de inform. pe unitate este mai mare decât pe exterior.
- dim. unui bloc depinde de sistemul de operare, pt Windows avem 512 octeți;
- volumul de date pe un hard-disk se poate determina;



- capetele se mișcă sincron
- mișcarea nu este radială, ci sector;

Pe o suprafață se găsește:

- un nr. de piste
- nr. de sectoare

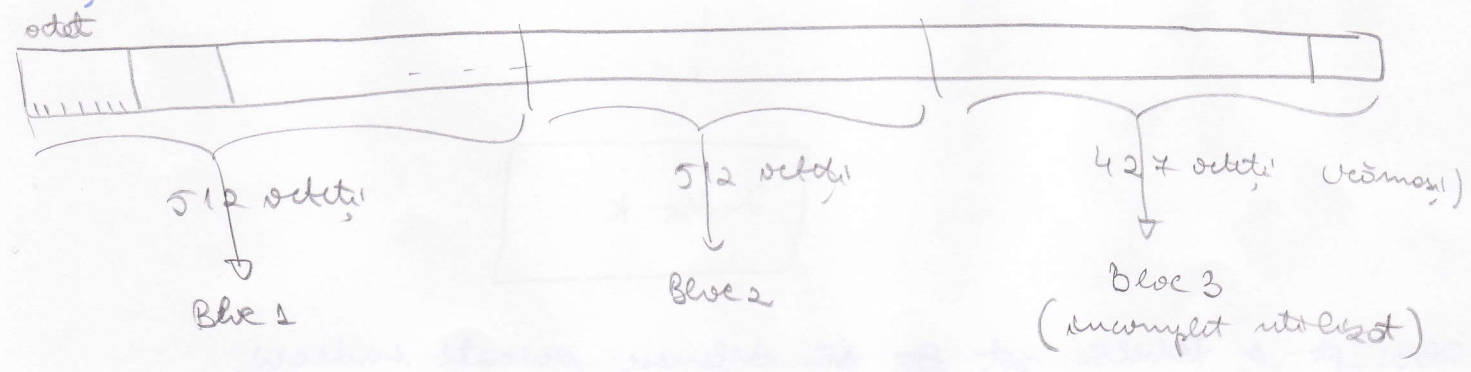
$$\left[\frac{\text{nr. piste} \times \text{nr. de sectoare} \times \text{nr. capete}}{2} \right] \times 512 \text{ octeți} \rightarrow \text{volumul de date pe hard pt Windows}$$

nr. de blocuri pe o suprafață.

Un cilindru reuneste toate pistele ^{cu} aceeași lășă; operația de poziționare a capetului cilindrului este consumatoare de timp.

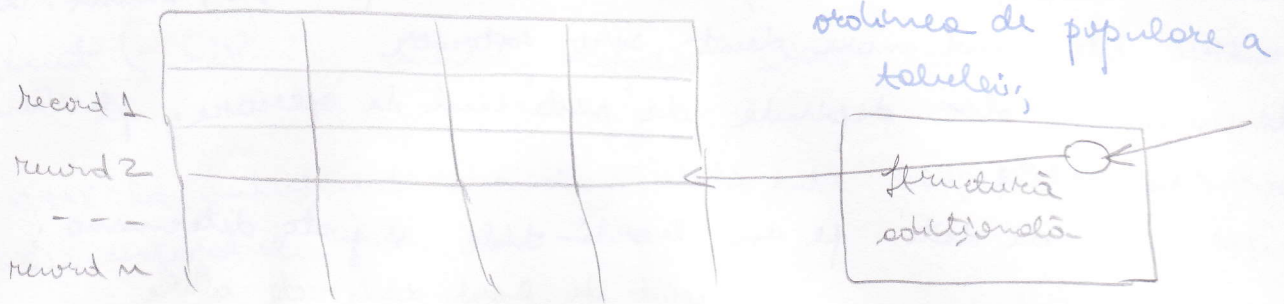
Pe pistă 0 se ține FAT (File Allocation Table) (siguranță mare); pistă 0 este cea de la exterior.

Fișier sequential = înălțare de octeți



- pt B3 sau End of File, pt a nu putea să mai adaug nimic.
- pt B1, B2, B3 - doresc să fie stocate contiguu a datelor (blocurile unele lângă altele);
- stocare contiguă = datele se fac găse pe același cilindru sau pe cilindri adiacenți (pt a face poziționarea capetului de citire ușor).

Ordinea fizică - tabele = ordinea stocării înregistrărilor;



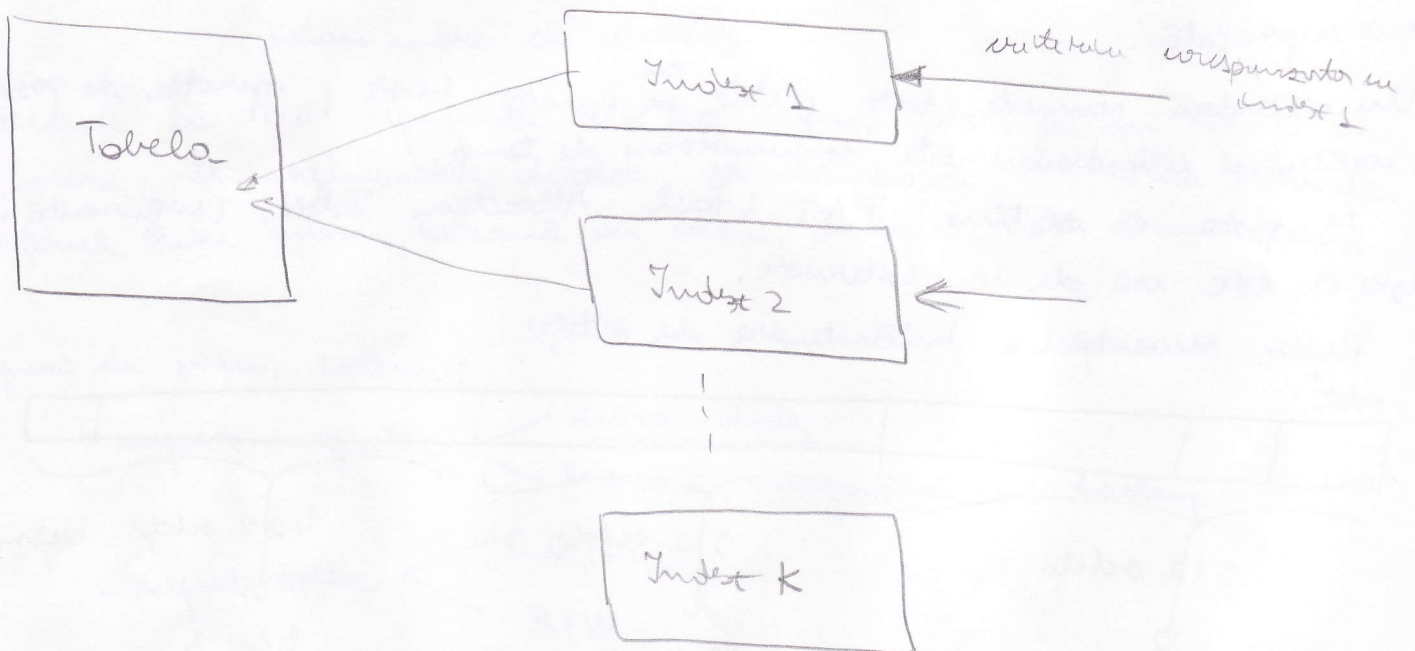
Ordinea fizică poate să coincidă sau nu cu ordinea în care se găsesc datele în tabelă → decizia portabilor face după un anumit comp.

Structura adițională hr. să aibă volum mai mic decât tabela propriu-zisă.

(Se numește mai interesantă blocul în care este poziționată o înregistrare)

INDEX: reprezintă o structură adițională de date asociată unei tabeli prin care se definește o ordine logică ce poate fi diferită de ordinea fizică.

O tabelă poate avea mai multe informații adiționale de tip index corespunzător criteriilor de acces.



OBS: pt o tabelă pot fi ~~fi~~ ~~de~~ definiți oricâți indici.

- în # operație de acces la tabelă se manipulează volumul de date din index.

clasificare index:

- a) index primar: este index la o tabelă sortată după valoarea câmpului cheie, cu indexare pe câmp cheie.
- b) index secundar: cazul tabelelor pt care accesul se face după câmpul cheie, dar tabela este neordonată după acest câmp.
- c) index de grup (cluster): cazul tabelelor în care indexarea și accesul se face după un câmp non-cheie;

$a + b + c \Rightarrow$ index de nivel 1;

\Rightarrow index multinivel: nr. de niveluri depinde de dim. tabelii; se poate implementa prin 2 tehnologii:

- a) arbori B
- b) arbori B+

obs: indexul multinivel se poate implementa numai dacă mediul de BD permite acest lucru.