

BAZE DE DATE

CONCEPTE ȘI PROBLEMATICĂ

În acest moment termenul de **bază de date** a intrat în limbajul curent fiind folosit uneori și în alte accepțiuni decât cea de proveniență. În acest capitol sunt definite conceptele de **bază de date** (BD) și **sistem de gestiune a bazelor de date** (SGBD) și se face o trecere în revistă a elementelor continute în aceste definiții. Sunt prezentate apoi **funcțiile** pe care trebuie să le asigure un SGBD și **categoriile de utilizatori** ale unui astfel de sistem. Capitolul se încheie cu prezentarea celor **trei nivele de descriere** a unei baze de date și a conceptului de **independența date-program**.

1.1. Bază de date și Sistem de gestiune a bazelor de date

Literatura de specialitate conține mai multe definiții pentru conceptele de **bază de date** și **sistem de gestiune a bazelor de date**. În continuare sunt prezentate cele din [DeAd82] pentru valoarea lor descriptivă.

Definiție: O bază de date este un ansamblu structurat de date înregistrat pe suporturi accesibile calculatorului pentru a satisface simultan cerințele mai multor utilizatori într-un mod selectiv și în timp util.

Definiție: Un sistem de gestiune a bazelor de date este ansamblul de programe care permit utilizatorului să interacționeze cu o bază de date.

Aceste definiții conțin majoritatea elementelor importante ale problematicei bazelor de date prezentate în capitolele următoare:

... un ansamblu structurat de date ...

Nu orice colecție de date este o bază de date. O cerință primordială este aceea a organizării acestora după anumite reguli. Regulile și conceptele care permit descrierea structurii unei BD formează **modelul datelor**. În timp au fost definite trei astfel de modele:

- **Modelul ierarhic** în care datele erau organizate sub forma unui arbore, nodurile constând din înregistrări iar arcele referințe către alte noduri.
- **Modelul rețea** în care datele erau organizate sub forma unui graf orientat. Nodurile și arcele au aceeași semnificație ca mai sus.
- **Modelul relațional** în care, intuitiv, datele sunt organizate sub forma de tabele.

Dezavantajul principal al primelor două modele constă în operarea greoie cu datele: orice operație necesită navigația printr-un arbore sau o rețea pentru localizarea datelor cautate

sau a locului în care se inserează noile înregistrari. De aceea ele nu se mai folosesc în realizarea sistemelor de gestiune moderne, acestea fiind bazate pe modelul relațional al datelor. Popularitatea acestui model este datorata simplității sale (din punct de vedere al utilizatorului) și a posibilității de definire a unor limbaje neprocedurale de descriere și manipulare a datelor.

Termenul de **relație** (care da denumirea modelului) provine din matematica iar reprezentarea intuitiva a unei relații este o tabela. În cazul modelului relațional descrierea structurii unei baze de date consta în principal din descrierea tabelelor componente: denumire, lista de coloane și tipul datelor din acestea.

De asemenea descrierea poate include informații despre elementele de identificare ale înregistrărilor dintr-o tabela (cheie primara), a coloanelor comune cu alte tabele (chei straine) și a altor restricții și verificari de corectitudine pentru datele stocate. Aceste informații sunt cunoscute sub numele generic de **constrangeri de integritate**.

Capitolul 3 prezinta bazele teoretice ale modelului relațional iar Capitolele 4 și 5 limbajul de cereri SQL care s-a impus ca standard pentru sistemele relaționale.

Operațiile de proiectare și descriere a structurii unei baze de date sunt facute în general cu ajutorul modelului **entitate-asociere**. Acesta este un model generic de descriere a datelor sub forma grafica - diagrame entitate asociere - din care se poate trece pe baza unor reguli de transformare în alte modele de date. Capitolul 2 prezinta doua versiuni ale acestui model iar în cadrul Capitolului 3 sunt descrise regulile de transformare din model entitate-asociere în model relațional.

...înregistrat pe suporturi accesibile calculatorului ...

Daca ansamblul de date nu este înregistrat pe suporturi accesibile calculatorului acesta nu se poate numi bază de date în acceptiunea lucrării de fata.

În limbajul curent se intalnesc expresii ca: "avem aceasta informație în baza noastra de date" și în cazurile în care datele respective sunt de exemplu stocate sub forma unor fise (pe hartie) sortate alfabetic sau dupa alte criterii. În acest caz este vorba despre o extensie a termenului de **bază de date**.

În cazul sistemelor de gestiune a bazelor de date suporturile pe care sunt stocate datele sunt în principal magnetice și optice.

... pentru a satisface simultan cerintele mai multor utilizatori ...

Funcțiile unui SGBD relative la accesul utilizatorilor la baza de date sunt urmatoarele:

1. Gestiunea utilizatorilor. Un SGBD trebuie sa permita crearea, modificarea și stergerea utilizatorilor. Operația este efectuata de obicei de administratorul bazei de date.

2. Concurenta la date. În cazul accesului simultan al mai multor utilizatori la aceleasi date un SGBD trebuie sa aiba mecanisme pentru a preintampina inconsistenta datelor. Iata un exemplu de caz:

Fie o portiune a bazei de date, notata A, în care se tine numarul de locuri disponibile la o cursa aeriana și doua executii ale unui program de rezervare rulat de doua agentii de bilete diferite. Daca programul este de tipul:

```
READ A      /* se citeste numarul de locuri disponibile din BD */  
A = A - 1    /* se decrementează cu o unitate */  
WRITE A     /* se scrie noul numar de locuri disponibile în BD */
```

atunci este posibil ca pașii celor două execuții să fie întretreși în felul următor:

Moment de timp	Agentia 1	Agentia 2	A în BD
t1	READ A		10
t2		READ A	10
t3	$A = A - 1$		10
t4		$A = A - 1$	10
t5	WRITE A		9
t6		WRITE A	9

Se observă că deși s-au rezervat 2 locuri, A a scăzut cu doar o unitate. O astfel de situație trebuie preintampinată.

Capitolul 6 prezintă problemele legate de blocarea (acapararea) unor porțiuni ale BD de către o execuție a unui program, rezolvarea problemelor pe care le poate ridica așteptarea circulară pentru deblocarea acestor porțiuni (deadlock), execuția pașilor programelor de actualizare a BD echivalentă cu o execuție secvențială a programelor (serializabilitate) și reguli de scriere a programelor de aplicație pentru rezolvarea problemelor de acces concurrent.

... într-un mod selectiv ...

Orice SGBD are mecanisme prin care diversilor utilizatori sau categorii de utilizatori li se asociază drepturi de acces specifice la obiectele bazei de date. În acest mod fiecărui utilizator i se da dreptul de a efectua doar operațiile specifice activității sale și doar pe acea porțiune a bazei de date care este necesară pentru acestea.

Mecanismul de drepturi de acces are ca obiective principale:

- Blocarea accesului unor categorii de utilizatori la date pe care **nu trebuie** să le acceseze. În acest fel este asigurată una dintre funcțiile de bază ale unui SGBD și anume **confidențialitatea datelor**.
- Blocarea accesului unor categorii de utilizatori la date de care **nu au nevoie** în activitatea lor, minimizându-se astfel riscul distrugerii accidentale a datelor prin operații necorespunzătoare.

Fiecare tip de SGBD are propriile sale mecanisme de descriere a drepturilor de acces bazate în principal pe acordarea sau neacordarea dreptului de a citi sau scrie diverse porțiuni ale bazei de date. O prezentare detaliată a acestei probleme se găsește în Capitolul 7.

... și în timp util ...

În cazul bazelor de date de dimensiuni mari este evident că orice căutare care s-ar baza pe o parcurgere secvențială a înregistrărilor din tabele ar duce la timpi de răspuns inadecvat de mari. De asemenea, operații mai complicate prin care se regăsesc date stocate în mai multe tabele legate între ele prin coloane comune pot duce în lipsa unor algoritmi specifici la timpi de execuție inacceptabili.

De aceea orice SGBD are mecanisme prin care minimizează timpul de răspuns, mecanisme bazate în special pe indecși și modalități specifice de organizare fizică a datelor.

1.2. Funcțiile unui SGBD

O definiție alternativa a conceptului de bază de date este urmatoarea [ElNa94]: o bază de date este o colecție de date interconectate. Deci ea reprezinta depozitul de date al oricarei aplicații de gestiune, partea sa statica. Operațiile asupra datelor sunt efectuate de sistemul de gestiune a bazelor de date. El este cel care asigura structurarea datelor, accesul concurrent al utilizatorilor, selectivitatea accesului și timpi de executie normali pentru cereri. Dar acestea sunt doar o parte din operațiile pe care acesta trebuie sa le asigure. Funcțiile unui sistem de gestiune a bazelor de date sunt urmatoarele:

1.2.1. Descrierea datelor

Un SGBD trebuie sa includă posibilitatea descrierii sturcturii obiectelor care formează baza de date. În cazul bazelor de date relaționale aceasta consta în principal în posibilitatea creerii și modificării structurii tabelelor și constrangerilor de integritate asociate acestora.

Limbajul prin care se realizează aceste operații se numeste Limbaj de Descriere a Datelor (LDD) și în cazul primelor sisteme de gestiune el era implementat sub forma unor module separate. În sistemele relaționale bazate pe SQL aceste operații au fost incluse în limbaj sub forma comenzilor de tip CREATE (pentru creare) sau ALTER (modificare).

O caracteristica importanta a sistemelor relaționale este aceea ca descrierea structurii bazei de date este stocata în acelasi mod ca și baza de date în sine. Exista pentru aceasta o serie de tabele gestionate direct de sistemul de gestiune numite **cataloge** care contin descrierea obiectelor care formează baza de date. Aceste tabele sunt accesate de sistem pentru a regasi informațiile necesare executiei cererilor dar pot fi accesate și de utilizatori pentru a obtine informații privind structura bazei de date. Astfel toate informațiile de sistem necesare unui SGBD relațional sunt stocate în acelasi mod, inclusiv descrierea utilizatorilor și a drepturilor de acces ale acestora.

Aceasta abordare a fost necesara pentru a nu avea mecanisme diferite de acces la datele de sistem și la datele stocate în baza de date.

Din cele de mai sus rezulta ca nu orice program care gestionează date este un SGBD. Sa luam exemplul unui program scris în limbajul C care gestionează niste fișiere de înregistrari a caror structura este cablata în program. Acesta nu respecta funcția de descriere a datelor necesara unui SGBD deoarece utilizatorul:

- Nu poate adauga noi fișiere de date.
- Nu poate modifica structura fișierelor de date gestionate (adaugari sau stergeri de campuri ale înregistrarilor fișierelor, schimbarea tipului acestora).
- Structura bazei de date este cablata în program și nu este stocata în baza de date.

Un SGBD trebuie sa permita deci definirea și modificarea de catre utilizatorii autorizați a structurii bazei de date, limitarile fiind doar de natura tehnica sau privind tipurile de obiecte care se pot înregistra: dimensiune maxima pentru o tabela, numar maxim de coloane sau linii dintr-o tabela, tipul unei coloane ales dintr-o multime predefinita de tipuri posibile, etc.

1.2.2. Utilizarea datelor

Aceasta funcție include operațiile de lucru cu datele înregistrate într-o bază de date. Cele patru categorii de operații principale sunt următoarele:

- Inserarea de noi date. Aceasta se concretizează prin adaugarea de noi linii în tabelele care formează baza de date.
- Stergerea de linii din tabele.
- Actualizarea datelor, însemnând modificarea conținutului unor linii existente în tabele.
- Regasirea datelor după anumite criterii de cautare

Pentru implementarea acestei funcții fiecare SGBD are un Limbaj de Manipulare a Datelor (LMD) care poate fi un modul separat sau inclus în limbajul sistemului cum este în cazul SQL.

Operația de regasire este esențială, ea fiind necesară și în cazul celorlalte trei operații:

- În cazul inserării, dacă aceeași informație nu poate fi prezentă de mai multe ori în baza de date, este necesară o cautare prealabilă pentru a verifica acest lucru.
- În cazul stingerii și actualizării este necesară localizarea datelor care vor fi eliminate sau modificate.

Pentru efectuarea unei operații de acest tip un SGBD execută mai mulți pași:

1. Consultarea cataloagelor de sistem pentru a verifica existența obiectelor implicate și a afla structura și proprietățile acestora precum și pentru a verifica dacă utilizatorul care a lansat comanda are dreptul de a efectua operația respectivă
2. Crearea unui plan de execuție optim pentru operația respectivă, luând în considerație existența unor eventuale structuri de cautare rapidă (indecsi).
3. Executia operației și verificarea faptului că rezultatul nu violează constrangerile de integritate existente
4. Executarea altor operații care pot fi declanșate automat de operația de bază (în cazul în care sistemul suportă astfel de operații).

Și din punct de vedere al acestei funcții nu toate programele de gestiune de date sunt SGBD. În cazul programului scris în limbajul C descris anterior, dacă operarea se face prin intermediul unui arbore de meniuri care permite efectuarea unui set prestabilit de operații funcția de utilizare a datelor nu este îndeplinită. Adăugarea unei noi opțiuni în program descriind o operație care nu a fost prevăzută nu se poate face de utilizator ci doar prin modificarea și recompilarea programului.

O situație specială este aceea a programelor dedicate care au însă la bază un sistem de gestiune a bazelor de date. În acest caz SGBD-ul dispune de toate facilitățile de descriere și utilizare a datelor dar acestea nu sunt puse la dispoziția utilizatorului aplicației decât în măsura în care au fost cuprinse în meniurile acesteia.

1.2.3. Integritatea datelor

Majoritatea sistemelor de gestiune permit definirea unor reguli pe care datele stocate trebuie sa le verifice numite **constrangeri de integritate**. În cazul în care o operație are ca rezultat violarea acestor restricții aceasta este automat rejectata și nu are efect în baza de date. În felul acesta este asigurata o mai mare siguranta în ceea ce priveste corectitudinea datelor.

Sa consideram o bază de date continând informații despre facultăți, studenții înscriși și notele obtinute de acestia. Impunerea unor constrangeri de integritate poate preveni în acest caz:

- Introducerea accidentala de valori incorecte în baza de date. De exemplu pentru tabela continând notele studentilor se poate defini o restricție privind intervalul de valori admisibil pentru acestea: 0-10. Orice operație de introducere a unei noi note sau de modificare a uneia existente care nu respecta acest interval de valori va genera o eroare și nu va fi dusa pana la capat.
- Lasarea necompletata a unor informații necesare. În cazul introducerii unui nou student sau a modificării informațiilor despre unul deja înregistrat, o restricție de acest tip asociata coloanei sau coloanelor pe care este memorat numele studentului va evita aparitia în baza de date a unor înregistrari în care numele este necompletat.
- Alocarea accidentala a aceluasi numar matricol pentru mai multi studenti în cazul în care pentru coloana continând acest numar s-ar defini o constrangere de unicitate a valorilor.
- Introducerea accidentala a unui student cu un cod incorect al facultății la care acesta este înscriș. Acest lucru se poate realiza definind o constrangere de integritate prin care se verifica existenta în tabela de facultăți a valorii respective inainte de scrierea ei în tabela de studenti.

Definirea de constrangeri de integritate nu previne inasa total erorile accidentale de operare: de exemplu introducerea din greseala a unei note de 4 în loc de 5 nu va fi semnalata, ambele valori fiind în intervalul admisibil.

1.2.4. Confidentialitatea datelor

În cazul unui SGBD accesul la date este permis doar utilizatorilor înregistrați și doar în masura drepturilor de acces alocate. Paragraful 1.3. contine o prezentare detaliata privind tipurile de utilizatori ai unui sistem de gestiune și rolurile acestora.

Un utilizator este identificat printr-un nume-utilizator și o parola. Fiecarui utilizator i se permite accesul doar la o portiune a bazei de date și doar pentru a efectua anumite tipuri de operații. În cazul bazelor de date accesate în retea se pot defini de asemenea locațiile de la care utilizatorul poate interacționa cu baza de date.

Toate aceste informații relative la ce, cum și de unde poate accesa datele un utilizator reprezinta drepturile de acces asociate acestuia și sunt stocate în cataloagele sistemului.

Asa cum s-a mentionat anterior, prin mecanismul de drepturi de acces este asigurata funcția de confidentialitate a datelor. Este prevenita astfel posibilitatea de acces la date a persoanelor care nu au acest drept: de exemplu, în cazul sistemului informatic al unei organizații, accesul la datele privind salariile angajaților, detalii ale proceselor tehnologice, datele contabile sau alte informații sensibile nu vor putea fi accesate decat de categorii restranse de utilizatori.

Din acest punct de vedere nu orice pachet de programe de gestiune a datelor poate fi numit SGBD deoarece unele dintre acestea fie nu contin nici un fel de facilitate privind gestiunea utilizatorilor și asigurarea confidentialității fie aceste facilități sunt rudimentare și nu pot fi considerate ca atare.

O alta posibilitate de asigurare a confidentialității este aceea a accesului la datele din baza de date doar prin intermediul unor programe de aplicație. Utilizatorii acestor programe nu sunt în același timp și utilizatori înregistrați ai SGBD-ului care gestionează datele iar porțiunea din baza de date la care au acces este cablata în program.

1.2.5. Accesul concurent la date

Există trei tipuri de probleme pe care un SGBD trebuie să le rezolve pentru a asigura accesul concurent corect al mai multor utilizatori la aceeași bază de date:

Facilități de blocare a unor porțiuni ale bazei de date. Aceasta înseamnă că o execuție a unui program poate capata un acces exclusiv la o porțiune a bazei de date, denumită și **articol**, moment în care orice altă execuție trebuie să aștepte deblocarea acesteia pentru a continua. Dacă notăm operațiile de blocare și deblocare cu **LOCK** și **UNLOCK**, programul de rezervare de locuri descris anterior este următorul:

```

LOCK  A      /* se blochează articolul A */
READ  A      /* se citește din BD numărul de locuri disponibile */
A =    A - 1   /* se decrementează cu o unitate */
WRITE A      /* se scrie noul număr de locuri disponibile în BD */
UNLOCK A     /* se deblochează articolul A */

```

În acest caz, prima dintre cele două execuții care va reuși să blocheze cu succes articolul A își va executa toți pașii înaintea celeilalte, nevoită să aștepte deblocarea lui A:

Moment de timp	Agentia 1	Agentia 2	A în BD
t1	LOCK A		10
t2	READ A	Așteptare	10
t3	A = A - 1	Așteptare	10
t4	WRITE A	Așteptare	9
t5	UNLOCK A	Așteptare	9
t6		LOCK A	9
t7		READ A	9
t8		A = A - 1	9
t9		WRITE A	8
t10		UNLOCK A	8

Fiecare tip de SGBD are propriul sau mecanism de definire a dimensiunii articolelor și a tipurilor de blocări care se pot efectua.

Modalități de evitare sau de eliminare a interblocării. Blocarea articolelor rezolvă o serie de probleme dar crează premisele apariției interblocajului, situație în care două execuții de programe așteaptă deblocarea unui articol blocat de cealaltă.

Să luăm cazul a două programe care au următoarea structură:

Program P1	Program P2
LOCK A	LOCK B
LOCK B	LOCK A
.
UNLOCK B	UNLOCK A
UNLOCK A	UNLOCK B

În cazul în care atât P1 cât și P2 au executat prima instrucțiune blocând articolele A și respectiv B, fiecare dintre ele intra în așteptare: P1 așteaptă ca P2 să-l deblocheze pe B iar P2 așteaptă ca P1 să-l deblocheze pe A.

O astfel de situație se numește **deadlock** și trebuie fie evitată fie eliminată de către SGBD.

Execuția serializabilă. În cazul mai multor execuții simultane care accesează baza de date se considera ca efectul lor este corect dacă rezultatul final este identic cu execuția lor succesivă. Această proprietate se numește **serializabilitate** și un SGBD trebuie să pună la dispoziție mecanisme de asigurare a serializabilității execuțiilor programelor de acces.

Capitolul 6 face o prezentare a bazelor teoretice ale accesului concurent la date în sistemele de gestiune a bazelor de date relaționale.

1.2.6. Siguranța în funcționare

Deși nu este legată direct de cele prezentate până acum, siguranța în funcționare este o caracteristică esențială pentru un SGBD și conține acele elemente care exclud sau minimizează posibilitatea de pierdere a datelor datorată incidentelor software sau hardware. Principalele facilități pe care un sistem de gestiune a bazelor de date trebuie să le asigure din acest punct de vedere sunt următoarele:

- **Salvarea datelor.** În cazul sistemelor mai vechi aceste facilități erau suplinite de opțiunile de copiere sau arhivare a fișierelor bazei de date oferite de sistemul de operare pe care rula SGBD-ul. Actualmente implementarea operațiilor de salvare este mult mai sofisticată având în vedere dificultatea teoretică și practică a efectuării de copii de siguranță consistente ale bazei de date în condițiile în care aplicația rulează non-stop și operarea nu poate fi oprită pentru efectuarea salvării.
- **Restaurarea după incident.** În cazul apariției de incidente hardware sau software care au ca efect distrugerea bazei de date este necesară efectuarea operației de restaurare care să minimizeze volumul de operații al căror efect se pierde. În acest scop cele mai multe tipuri de SGBD au posibilitatea de a înregistra toate schimbările aparute în baza de date sub forma unor **fișiere jurnal**. În acest fel operația de restaurare după incident se face restaurând ultima copie de siguranță și reefectuând toate prelucrările consistente înregistrate în jurnal după momentul în care a fost efectuată copia.

Aceste facilități sunt necesare mai ales în cazul aplicațiilor care gestionează date importante și pentru care pierderea acestora ar avea implicații deosebite, de exemplu în cazul aplicațiilor bancare.

Eficiența acestor facilități poate fi foarte importantă în momentul demarării unei noi aplicații și actualmente cele mai bine vândute tipuri de SGBD sunt tocmai cele care au mecanisme performante de asigurare a siguranței în funcționare.

1.3. Categoriile de utilizatori

În paragrafele precedente a fost folosit frecvent termenul de **utilizator**. În cele ce urmează sunt prezentate categoriile de utilizatori care interacționează cu o bază de date.

Din punct de vedere al **drepturilor de acces**, ca și în cazul sistemelor de operare, un SGBD are două tipuri principale de utilizatori:

1.3.1. Utilizatori privilegiați

Aceștia sunt utilizatori care au dreptul de a efectua toate tipurile de operații puse la dispoziție de sistem. Termenul generic pentru acest tip de utilizatori este cel de **administrator al bazei de date** și în general este vorba de una sau mai multe persoane care răspund de buna funcționare a SGBD-ului. Principalele funcții îndeplinite de administrator sunt:

- Crearea și modificarea structurii bazei de date. Aici sunt incluse operațiile de creare și modificare a tabelor și a celorlalte obiecte ale bazei de date necesare aplicațiilor folosite în organizația respectivă.
- Gestiunea utilizatorilor. Administratorul este cel care creează noi utilizatori pentru SGBD, conferă drepturile de acces la date și poate schimba acest tip de informații pentru orice utilizator. El răspunde deci și de securitatea datelor și prevenirea accesului neautorizat la baza de date în întregime sau la porțiunile sale care nu sunt publice.
- Modificarea parametrilor SGBD. În cazul sistemelor de gestiune complexe performanțele diferitelor tipuri de operații sunt influențate de o serie de parametri la care un utilizator obișnuit nu are acces. Administratorul este cel care monitorizează performanțele sistemului și poate schimba parametrii acestuia pentru optimizarea anumitor operații.
- Salvarea periodică a bazei de date (crearea de copii de siguranță) și restaurarea sa în caz de incident software sau hardware.
- Poate transfera unele dintre drepturile sale de acces și altor utilizatori care nu sunt administratori ai bazei de date în cazurile în care acest lucru este necesar dar răspunde de eventualele breșe de securitate care pot apărea fiind cel mai în măsură să evalueze problemele ridicate de un astfel de transfer.

1.3.2. Utilizatori neprivilegiați

Aceștia sunt utilizatorii obișnuiți ai SGBD-ului și dispun de drepturile de acces care le-au fost alocate de administratorul bazei de date.

Majoritatea sistemelor de gestiune permit definirea de categorii generice de utilizatori (numite roluri) iar fiecare utilizator individual are asociat unul sau mai multe roluri, moștenind drepturile de acces ale acestora. Este ușurată astfel operația de creare a unui nou utilizator.

Există două tipuri de utilizatori neprivilegiați:

- Utilizatorul este o persoană fizică lucrând direct cu SGBD-ul prin intermediul interfețelor pe care acesta le pune la dispoziție. De exemplu în Oracle o astfel de

interfata este programul **sqlplus** prin care se pot executa cereri SQL asupra bazei de date.

- Utilizatorul este o entitate generica având asociat un nume-utilizator și o parola prin intermediul careia o multime de programe de aplicație se conectează la SGBD pentru a efectua diverse operații. Persoanele care utilizează aceste aplicații nu cunosc numele-utilizator sau parola folosite pentru accesul la date, existând eventual mecanisme paralele de autentificare a utilizatorilor autorizați ai aplicației.

Din punct de vedere al relației cu una dintre bazele de date gestionate de un SGBD persoanele care accesează datele se împart în doua categorii:

Utilizatori care proiectează, realizează și intretin aplicația. În aceasta categorie intra toti cei implicați în crearea unei aplicații și în activitatea de mentenanta și adaptare la nevoile beneficiarilor. Activitățile efectuate de acestia sunt:

- Analiza de sistem și proiectarea bazei de date și a modulelor de program ale aplicației.
- Dezvoltarea programele conform proiectului rezultat din analiza de sistem
- Asigura suportul pentru utilizarea aplicației. În multe sisteme de gestiune depanarea incidentelor și intretinerea bazei de date se pot face de la distanta de personalul companiei care a livrat sau distribuit aplicația.

Acesti utilizatori pot primi de la administratorul bazei de date o serie de drepturi specifice acestuia: crearea de noi tabele, modificarea structurii celor existente, gestiunea constrangerilor de integritate și a altor obiecte din baza de date.

Utilizatori obisnuiti. Sunt cei care au acces la baza de date prin intermediul interfetelor pentru care sunt autorizați. Acesti utilizatori în general nu au dreptul de a modifica structura bazei de date a aplicației dar în anumite cazuri pot crea obiecte suplimentare pentru care dispun de drepturi nelimitate: creare, modificare, stergere și transfer al drepturilor de acces la acestea catre alti utilizatori.

1.4. Nivele de descriere a unei baze de date

O aceeași bază de date poate fi privita din diverse perspective rezultând descrieri diferite. Termenul consacrat pentru descrierea structurii unei baze de date este acela de **schema**. În literatura de specialitate exista o clasificare pe trei nivele a acestor descrieri: fizic, conceptual și extern. Gestionarea schemelor bazei de date pentru fiecare nivel este una dintre atribuțiile administratorului bazei de date.

1.4.1. Nivelul fizic

La acest nivel baza de date este descrisa din perspectiva stocarii sale pe dispozitivele fizice: identificarea discurilor și a cailor unde este stocata, numele fișierelor care formează baza de date, structura fizica a acestora, etc. Descrierea bazei de date la acest nivel poarta numele de **schema fizica** și sistemul de gestiune a bazelor de date pune la dispozitie facilitățile pentru înregistrarea și modificarea acesteia. Fiecare SGBD are în general asociat un model specific de descriere la nivel fizic a bazei de date.

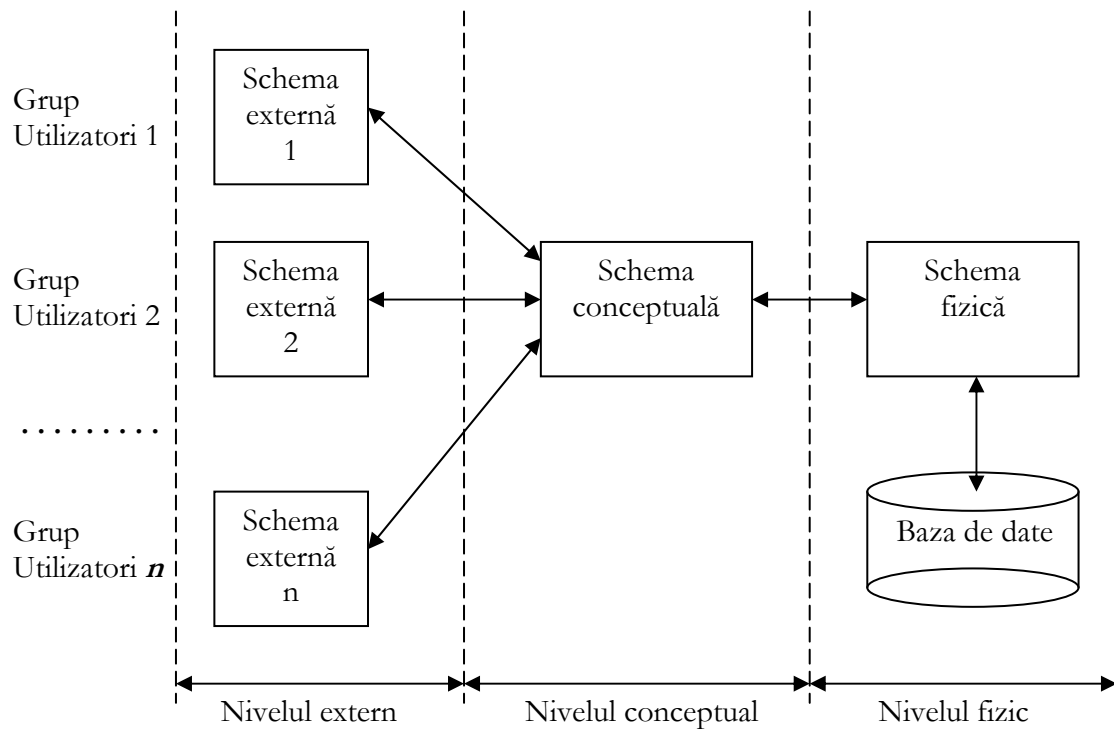


Fig. 1.1. Cele trei nivele de descriere ale unei baze de date

1.4.2. Nivelul conceptual

Descrierea bazei de date la acest nivel poartă numele de **schema conceptuală** (numită uneori și schema logică) a bazei de date. Ea constă într-o descriere abstractă dar exactă a structurii acesteia, lasând la o parte detaliile fizice de implementare.

Schema conceptuală este făcută în termenii modelului de date utilizat. Astfel, în cazul adoptării modelului relațional, aceasta constă în:

- Tabelele care formează baza de date
- Structura (coloanele) fiecărei table
- Tipul de date asociat coloanelor
- Elementele pe baza cărora se realizează interconectarea tabelor (coloane comune)
- Constrângeri de integritate
- Operații declanșate automat la modificarea unor elemente ale bazei de date

Implementarea schemei conceptuale se face cu ajutorul limbajului pentru descrierea datelor (LDD) asociat sistemului de gestiune utilizat.

1.4.3. Nivelul extern

Diferitele categorii de utilizatori ai unei baze de date au nevoie în activitatea lor doar de porțiuni specifice ale acesteia. Descrierea acestor porțiuni poartă numele de **scheme externe**. O bază de date are deci asociată o singură schemă fizică și o singură schemă conceptuală dar mai multe scheme externe.

Schemele externe sunt descrise de obicei cu ajutorul modelului de date folosit pentru schema conceptuala. În plus se specifica modul în care se face corespondenta între obiectele celor doua descrieri.

Daca pentru administratorului bazei de date schema externa coincide cu schema conceptuala, celelalte categorii de utilizatori accesează baza de date doar prin intermediul schemelor externe specifice acestora.

Din aceasta cauza, orice prelucrare lansata de un utilizator este translatata de catre SGBD mai intai la nivel conceptual și apoi la nivel fizic. Aceste operații pot fi consumatoare de timp. De aceea nu toate sistemele de gestiune dispun de mecanisme de descriere pentru fiecare dintre cele trei nivele:

- În cazul multor pachete de programe pentru calculatoare personale descrierea schemei fizice se face cu ajutorul LDD, o data cu schema conceptuala. De asemenea facilitățile pentru definirea schemelor externe sunt reduse sau inexistente.
- În cazul sistemelor de gestiune a bazelor de date complexe, descrierea schemelor externe se face prin definirea de vederi utilizator. O astfel de vedere este în cazul modelului relațional o tabela virtuala care nu exista la nivel fizic și conceptual în baza de date dar care se poate calcula pe baza tabelelor continute în aceasta ori de cate ori este folosita de operațiile lansate de utilizator.

1.4.4. Independenta datelor

Existenta celor trei nivele de descriere permite definirea conceptului de independență între datele stocate în baza de date și aplicațiile care utilizează aceste date.

Conceptul de independenta a datelor a aparut o data cu dezvoltarea sistemelor complexe de aplicații pentru care cablarea informațiilor structurale în program constituie o bariera în calea dezvoltării și modificării acestora. În lumea reala orice operație de modificare a bazei de date a unei aplicații se masoara și prin prisma costurilor materiale necesare modificării programelor care o folosesc. Minimizarea acestora a facut necesara implementarea facilităților de independenta a datelor în sistemele de gestiune a bazelor de date comerciale.

Exista doua tipuri de independenta:

Independenta logica reprezinta posibilitatea de schimbare a schemei conceptuale a bazei de date fara modificarea schemelor externe. Conditia este ca modificarea sa nu elimine nici unul dintre elementele necesare translației de la schema externa la schema conceptuala.

Operații ca:

- Adaugarea de noi tabele în baza de date
- Adaugarea de noi coloane în tabelele existente
- Modificarea numelor tabelor și coloanelor existente
- Adaugarea de noi constrangeri de integritate
- Modificarea în anumite limite a tipului datelor stocate în baza de date
- Restructurarea bazei de date prin mutarea unor coloane în alte tabele, fragmentarea unei tabele sau reunirea mai multor tabele.

pot fi suportate doar prin modificarea definiției schemelor externe, fara ca acestea sa sufere modificari structurale. În acest caz deoarece programele de aplicație și utilizatorii accesează baza de date doar prin intermediul schemelor externe modificarile schemei conceptuale nu vor fi sesizate de catre acestia.

Exemplu: Sa consideram o bază de date continând la nivel conceptual o tabela cu date despre studenti cu urmatoarea structura:

Student(CodStudent, Nume, CodFacultate, Medie)

și n scheme externe continând tabelele virtuale ***Student-1, Student-2, ..., Student-n*** definite astfel:

Student-i(CodStudent, Nume, CodFacultate, Medie) = liniile din tabela ***Student*** care pe coloana CodFacultate contin valoarea i .

În cazul modificării bazei de date prin adaugarea unei noi tabele cuprinzând lista specializarilor din cadrul universității și a unei noi coloane în tabela ***Student*** pentru a specifica la ce specializare este inscris fiecare student, tabelele cu datele studentilor din schemele externe vor ramane aceleasi daca le schimbam definiția:

Baza de date conceptuala:

Student(CodStudent, Nume, CodFacultate, Medie, CodSpecializare)
Specilizare(CodSpecializare, NumeSpecializare)

Schemele externe:

Student-i(CodStudent, Nume, CodFacultate, Medie) = valorile de pe coloanele CodStudent, Nume, CodFacultate și Medie din liniile din tabela ***Student*** care pe coloana CodFacultate contin valoarea i .

Independenta logice implica folosirea de catre SGBD a informațiilor de definiție a schemelor externe stocate în cataloagele sistemului pentru conversia oricarei operații din termenii schemei externe a aplicației sau utilizatorului care a lansat-o în termenii schemei conceptuale a bazei de date.

Independența fizică reprezinta posibilitatea de schimbare a schemei fizice a bazei de date fara modificarea schemei conceptuale și implicit a schemelor externe. Aceasta da posibilitatea reorganizarii fizice a bazei de date fara afectarea aplicațiilor care o folosesc. Operații ca:

- Schimbarea dispozitivelor fizice pe care este stocata baza de date
- Schimbarea numelor fișierelor fizice în care este stocata baza de date sau a directoarelor unde acestea sunt plasate
- Adaugarea de noi structuri de cautare rapida (indecsi) pentru cresterea vitezei de executie a anumitor operații
- Schimbarea în anumite conditii a structurii fizice a fișierelor bazei de date
- Schimbarea unor parametri ai sistemului de gestiune care afectează modul în care datele sunt stocate la nivel fizic

pot fi suportate doar prin modificarea cataloagelor sau a fișierelor de configurație pe care SGBD-ul le foloseste pentru a face translația de la schema conceptuala la schema fizica.

Cum nu toate sistemele de gestiune a bazelor de date implementează total cele trei nivele de descriere, posibilitatea de a asigura cele doua tipuri de independenta este conditionata de facilitățile oferite de sistem. În cazul posibilității asigurării independentei datelor avantajul consta în existenta premiselor scrierii de aplicații care pot fi usor dezvoltate, intretinute și actualizate.