

## **Baze de date online**

**Descrierea CIP a Bibliotecii Naționale a României**

**RĂDULESCU, FLORIN**

**Baze de date online** / Florin Rădulescu, Alexandru Boicea. –  
București : Editura Academiei Oamenilor de Știință din România,  
2011

Bibliogr.

Index

ISBN 978-606-8371-13-9

I. Boicea, Alexandru

004.65

**Editura Academiei Oamenilor de Știință din România**

**Adresa:** Splaiul Independenței, nr. 54, sectorul 5, cod 050094 București, România

**Redactor:** ing. Mihail CĂRUȚAȘU

**Documentarist:** ing. Ioan BALINT

**Coperta:** ing. sist. Adrian Nicolae STAN

**Copyright © Editura Academiei Oamenilor de Știință din România,  
București, 2011**

**Florin Rădulescu**  
**Alexandru Boicea**

# **Baze de date online**



**Editura Academiei Oamenilor de Știință din România**

**București**

**2011**



# PREFATĂ

Istoria bazelor de date relaționale începe în 1970 odată cu publicarea articolului lui Edgar Frank Codd „*A Relational Model of Data for Large Shared Data Banks*” care descrie modelarea datelor sub formă de relații (termen matematic, reprezentarea intuitivă a unei relații fiind o tabelă) și operațiile de bază cu acestea.

A durat aproape 10 ani până la apariția primelor sisteme de gestiune a bazelor de date (SGBD) bazate pe acest model, timp în care s-au dezvoltat algoritmi și metode de a realiza în timp util operațiile cu tabele.

O serie de firme prestigioase au investit în cercetări privind modelul relațional al datelor, printre acestea numărându-se și IBM care a lansat în anii '70 dezvoltarea unui prototip de cercetare numit System R. În cadrul acestui sistem – care nu a fost niciodată comercializat ca atare, fiind folosit doar ca instrument de cercetare – a fost dezvoltat și un limbaj de cereri prin care utilizatorul interacționa cu datele din baza de date numit SEQUEL (de la *Structured English-like Query Language*) redenumit apoi din motive comerciale SQL. System R a fost tatăl sistemului DB2 pe care IBM îl comercializează și în momentul actual.

Lucrarea de față își propune să prezinte atât fundamentele teoretice ale modelului relațional (capitolele 1-4 și 6) cât și limbajul de cereri SQL care a devenit în timp limbajul standard de comunicare cu un sistem de gestiune (capitolul 5).

În exemplele din capitolul dedicat SQL s-au folosit următoarele convenții care să asigure o mai mare claritate și lizibilitate:

- S-a preferat scrierea cu litere mari a cuvintelor cheie și identificatorilor.
- Fiecare nouă clauză a unei cereri a fost scrisă începând cu o nouă linie.
- S-a folosit indentarea în cazul expresiilor lungi și a subcererilor.



# CUPRINS

<b>1. Concepte și problematică .....</b>	<b>9</b>
1.1. Bază de date și Sistem de gestiune a bazelor de date .....	9
1.2. Funcțiile unui SGBD .....	11
1.3. Categoriile de utilizatori .....	13
1.4. Nivelele de descriere a unei baze de date .....	13
<b>2. Modelarea datelor .....</b>	<b>15</b>
2.1. Etapele dezvoltării unei aplicații .....	15
2.2. Modelul entitate-asociere clasic .....	18
2.3. Modelul entitate-asociere utilizat în instrumente CASE .....	25
<b>3. Modelul relațional .....</b>	<b>27</b>
3.1. Modele de date: ierarhic, rețea, relațional .....	27
3.2. Elementele de bază ale modelului relațional .....	28
3.3. Transformarea diagramelor EA în modelul relațional .....	30
3.4. Algebra relațională .....	31
3.5. Calculul relațional .....	36
<b>4. Proiectarea bazelor de date .....</b>	<b>37</b>
4.1. Anomaliile care apar în cazul proiectării incorecte .....	37
4.2. Dependente funcționale .....	38
4.3. Forme normale .....	45
4.4. Descompunerea schemelor de relație .....	47
4.5. Dependente multivaloarea. Forma normală 4 .....	52
<b>5. Limbajul SQL .....</b>	<b>55</b>
5.1. Interogarea datelor .....	56
5.2. Cereri simple .....	57
5.3. Cereri cu clauza WHERE .....	58
5.4. Metode de join .....	61
5.5. Equi-join și non equi-join .....	61
5.6. Joinul unei tabele cu ea însăși .....	62
5.7. Join extern .....	63
5.8. Join vertical .....	63
5.9. Funcții în SQL .....	65
5.10. Subcereri SQL .....	72
5.11. Crearea și definirea structurilor tabelare .....	75

5.12. Constrângeri de integritate .....	78
5.13. Comanda ALTER TABLE .....	81
5.14. Comanda INSERT .....	83
5.15. Comanda DELETE .....	83
5.16. Comanda UPDATE .....	84
<b>6. Tranzacții și acces concurent .....</b>	<b>85</b>
6.1. Prezentarea problematicei. Terminologie .....	85
6.2. Gestiunea tranzacțiilor .....	86
6.3. Serializabilitate .....	88
6.4. Asigurarea consistenței la citire .....	90
6.5. Protocolul de blocare în două faze .....	91



# 1. 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 conținute 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 ale unei baze de date și a conceptului de independență 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 două definiții apreciate 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 problematicii 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 acest moment majoritatea sistemelor de gestiune sunt bazate pe modelul relațional al datelor în care, intuitiv, datele sunt organizate sub formă de tabele.

Popularitatea acestui model este datorată 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 dă denumirea modelului) provine din matematică iar reprezentarea intuitivă a unei relații este o tabelă. În cazul modelului relațional descrierea structurii unei baze de date constă în principal din descrierea tabelelor componente: denumire, listă de coloane și tipul datelor din acestea.

### **...înregistrat pe suporturi accesibile calculatorului ...**

Dacă ansamblul de date nu este înregistrat pe suporturi accesibile calculatorului acesta nu se poate numi bază de date în accepțiunea lucrării de față.

În limbajul curent se întâlnesc expresii ca: "avem această informație în baza noastră de date" și în cazurile în care datele respective sunt de exemplu stocate sub forma unor fișe (pe hârtie) sortate alfabetic sau după 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 cerințele mai multor utilizatori**

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

1. Gestiunea utilizatorilor. Un SGBD trebuie să permită crearea, modificarea și stergerea utilizatorilor. Operația este efectuată de obicei de administratorul bazei de date.

2. Concurența la date. În cazul accesului simultan al mai multor utilizatori la aceleași date un SGBD trebuie să aibă mecanisme pentru a preîntâmpina inconsistența datelor.

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 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 diverșilor utilizatori sau categorii de utilizatori li se asociază drepturi de acces specifice la obiectele bazei de date. În acest mod fiecărui utilizator i se dă 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 funcțiunile 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.

### ... și în timp util ...

În cazul bazelor de date de dimensiuni mari este evident ca orice cautare 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 alternativă a conceptului de bază de date este următoarea: *o bază de date este o colecție de date interconectate*. Deci ea reprezintă depozitul de date al oricărei aplicații de gestiune, partea sa statică. Operațiile asupra datelor sunt efectuate de sistemul de gestiune a bazelor de date. El este cel care asigură structurarea datelor, accesul concurrent al utilizatorilor, selectivitatea accesului și timpi de execuție normali pentru cereri. Dar acestea sunt doar o parte din operațiile pe care acesta trebuie să le asigure. Funcțiile unui sistem de gestiune a bazelor de date sunt următoarele:

### Descrierea datelor

Un SGBD trebuie să includă posibilitatea descrierii structurii obiectelor care formează baza de date. În cazul bazelor de date relaționale aceasta constă în principal în posibilitatea creării și modificării structurii tabelor și constrângerilor de integritate asociate acestora.

Limbajul prin care se realizează aceste operații se numește 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).

### Utilizarea datelor

Această 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 adăugarea de noi linii în tabelele care formează baza de date.
- Ștergerea de linii din tabele.
- Actualizarea datelor, însemnând modificarea conținutului unor linii existente în tabele.
- Regăsirea datelor după anumite criterii de căutare.

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.

### **Integritatea datelor**

Majoritatea sistemelor de gestiune permit definirea unor reguli pe care datele stocate trebuie să verifice numite constrângeri de integritate. În cazul în care o operație are ca rezultat violarea acestor restricții aceasta este automat rejectată și nu are efect în baza de date. În felul acesta este asigurată o mai mare siguranță în ceea ce privește corectitudinea datelor.

Definirea de constrângeri de integritate nu previne însă total erorile accidentale de operare: de exemplu introducerea din greșeală a unei note de 4 în loc de 5 nu va fi semnalată, ambele valori fiind în intervalul admisibil.

### **Confidențialitatea datelor**

În cazul unui SGBD accesul la date este permis doar utilizatorilor înregistrați și doar în măsura drepturilor de acces alocate. Un utilizator este identificat uzual printr-un *nume-utilizator* și o *parolă*. Fiecărui utilizator i se permite accesul doar la o porțiune a bazei de date și doar pentru a efectua anumite tipuri de operații. În cazul bazelor de date accesate în rețea se pot defini de asemenea locațiile de la care utilizatorul poate interacționa cu baza de date.

O altă posibilitate de asigurare a confidențialității 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 cablată în program.

### **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:

1. *Facilități de blocare a unor porțiuni ale bazei de date.* Aceasta înseamnă că o execuție a unui program poate căpăta un acces exclusiv la o porțiune a bazei de date.
2. *Modalități de evitare sau de eliminare a interblocării.*
3. *Execuția serializabilă.* În cazul mai multor execuții simultane care accesează baza de date se consideră că efectul lor este corect dacă rezultatul final este identic cu execuția lor succesivă.

### **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 caror efect se pierde.

### 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:

#### Utilizatori privilegiați

Aceștia sunt utilizatori care au dreptul de a afecta toate tipurile de operații puse la dispoziție de sistem. Termenul generic pentru acest tip de utilizatori este cel de administratori ai bazei de date și în general este vorba de una sau mai multe persoane care răspund de buna funcționare a SGBD-ului.

#### 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.

### 1.4. Nivele de descriere pentru o bază de date

O aceeași bază de date poate fi privită din diverse perspective rezultând descrieri diferite. Termenul consacrat pentru descrierea structurii unei baze de date este acela de schema. În literatura de specialitate există o clasificare pe trei nivele a acestor descrieri: fizic, conceptual și extern.

### **Nivelul fizic**

Schema fizică este descrierea bazei de date din perspectiva stocării sale pe dispozitivele fizice: identificarea discurilor și a căilor unde este stocată, numele fișierelor care formează baza de date, structura fizică a acestora etc.

### **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.

### **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 asociate o singură schemă fizică și o singură schemă conceptuală dar mai multe scheme externe.

### **Independența datelor**

Existența 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 independență a datelor a apărut odată cu dezvoltarea sistemelor complexe de aplicații pentru care cablarea informațiilor structurale în program constituie o barieră în calea dezvoltării și modificării acestora. În lumea reală orice operație de modificare a bazei de date a unei aplicații se măsoară și prin prisma costurilor materiale necesare modificării programelor care o folosesc. Există două tipuri de independență:

**Independența logică** reprezintă posibilitatea de schimbare a schemei conceptuale a bazei de date fără modificarea schemelor externe. Condiția este ca modificarea să nu elimine nici unul dintre elementele necesare translației de la schema externă la schema conceptuală.

**Independența fizică** reprezintă posibilitatea de schimbare a schemei fizice a bazei de date fără modificarea schemei conceptuale și implicit a schemelor externe. Aceasta dă posibilitatea reorganizării fizice a bazei de date fără afectarea aplicațiilor care o folosesc.