

PARTEA I

BAZE DE DATE

Capitolul

GESTIUNEA DATELOR

1

1.1. Introducere: definiții, terminologie

Apariția și dezvoltarea calculatoarelor electronice au condus la amplificarea activităților legate de stocarea, interogarea și administrarea colecțiilor de date.

Astăzi, cele mai multe dintre activitățile noastre zilnice necesită accesarea și actualizarea informației dintr-o bază de date: extragerea unei sume de bani din contul bancar, rezervarea unei camere de hotel, cumpărarea unui bilet de avion, împrumutul unei cărți de la bibliotecă, plata facturilor de telefon, curent electric etc. Toate acestea se pot face rapid și în siguranță, pentru că datele respective sunt bine organizate într-o **bază de date** și administrate de un **sistem de gestiune a bazelor de date**.



Definiție

Bază de date (BD) =

= o colecție de date aflate în relație unele cu altele și structurată astfel încât să poată servi unui anumit scop.

= este un set de date corelate și organizate în scopul prelucrării lor rapide și concomitente de către mai multe persoane.

Exemple:

1. Baza de date a unui liceu, în care sunt înregistrate și actualizate permanent informații despre elevi, profesori, planuri de învățământ, baza materială etc.
2. Baza de date a unui muzeu, în care sunt înregistrate operele de artă (grupate după tip, autor, tehnică de lucru) și expozițiile itinerante (descrise prin perioadă, itinerariu, responsabil, custozii participanți).
3. Baza de date a unui magazin de muzică, în care sunt înregistrate albumele de muzică în funcție de tipul de suport fizic (CD, casetă etc.), stil, autori, soliști, anul apariției etc.

Info

Termenul "*Database*" (bază de date, în limba engleză) a apărut pentru prima dată în titlul unei conferințe organizate la Santa Monica, SUA, în 1964: *Development and Management of Computer Centered DataBase*.

Definiție

Sistem de gestiune a bazelor de date (SGBD) = un ansamblu de programe care permit crearea și administrarea unei baze de date. Prin urmare, un SGBD (*Database Management System*) este un pachet software de nivel înalt, care permite proiectarea, construirea și administrarea bazelor de date dedicate rezolvării problemelor din cele mai variate domenii ale vieții reale.



Exemple:

IMS, DB2 (până la **DB9**, de la IBM), **Ingres II** (de la Computer Associates International Inc.), **Oracle 10g** (de la Oracle Corporation), **Ms Access** (studiat în clasa a X-a), **FoxPro** (de la Microsoft), **Paradox**, **Visual dBase** (de la Borland), **Sybase Adapted Server** (de la Sybase Inc.), **IRIS** (de la Hewlett-Packard).

Definiții

A proiecta o bază de date înseamnă a-i stabili structura, adică elementele componente, caracteristicile acestora, restricțiile pe care trebuie să le respecte, relațiile dintre ele.



A construi o bază de date înseamnă a memora (a introduce) datele în baza de date, după proiectarea acesteia.

A administra o bază de date înseamnă a asigura: (1) accesul la date al utilizatorilor, în funcție de drepturile fiecăruia (funcția și importanța fiecăruia în organizația respectivă), (2) coerența bazei de date (recuperarea informației atunci când au loc incidente: întreruperea alimentării cu curent electric, operații contradictorii etc.), (3) securitatea datelor stocate etc.

A interoga o bază de date înseamnă a extrage și a vizualiza datele care îndeplinesc anumite criterii și condiții.

A actualiza o bază de date înseamnă a modifica structura sau informațiile stocate în baza de date.

✓ **Atenție!**

Nu orice colecție de date este o bază de date.



De exemplu, lista cărților dintr-o bibliotecă NU este o bază de date, ci un simplu inventar de obiecte, **o listă, un tabel**. Dacă la această listă adăugăm lista cititorilor abonați la bibliotecă (eventual și lista angajaților bibliotecii) și luăm în considerare activitățile specifice bibliotecii (achiziționarea de cărți noi, împrumutul și restituirea cărților de către cititori, eventual administrarea sarcinilor de serviciu ale angajaților), atunci avem de a face cu o bază de date.

Prin urmare, față de un inventar (un tabel), o bază de date are următoarele proprietăți:

- ◆ reprezintă un anumit aspect al lumii reale, numit *microuniversul bazei de date*; orice modificare care se produce în acest microunivers se reflectă în baza de date (de exemplu: cumpărarea unei noi casete în vederea închirierii, modificarea diferenței permise între cursul de cumpărare și cel de vânzare al valutei etc.);
- ◆ este o colecție de date coerentă din punct de vedere logic și având un înțeles intrinsec (de exemplu: din baza de date asociată bibliotecii liceului nu vor face parte cărțile de telefon sau lista de materiale didactice din laboratorul de chimie);
- ◆ este proiectată, construită și administrată, având permanent în vedere un anumit scop; o bază de date este destinată utilizării de către un anumit grup de persoane și permite efectuarea unui anumit set de operații.



EXERCITIU

Bază de date sau tabele?

Dați câteva exemple de situații în care înregistrarea și administrarea datelor necesită folosirea unui tabel, respectiv a unei baze de date.

1.2. Date și informații

Am definit termenul de bază de date, dar nu am definit termenul de dată și nici pe cel de informație.



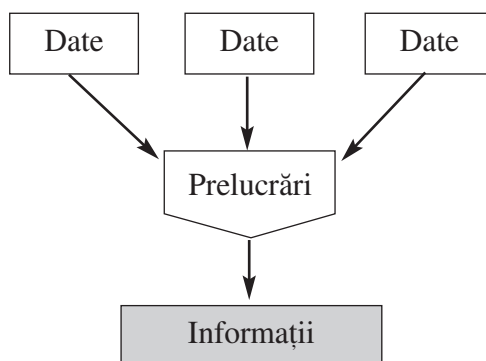
Definiție

Date =

= elemente, fapte, fenomene, procese etc. cunoscute, care pot fi culese și înregistrate și care au un înțeles implicit.

= fiecare dintre numerele, mărimile, relațiile etc. care servesc pentru rezolvarea unei probleme sau care sunt obținute în urma unei cercetări și urmează să fie supuse unei prelucrări (cf. DEX).

Cuvântul „dată” este de origine latină și provine de la verbul *a da*. În limba engleză, substantivul **dată** (*date*, la plural) se traduce prin **datum** (*data*, la plural). Exemple de date sunt: cantitățile de mere obținute anual într-o livadă de pomi fructiferi, activitățile turistice propuse de ghid participanților la o excursie; modificările climatice suferite de o regiune a globului terestru de-a lungul unui număr de ani, cursul bancar al unei valute de-a lungul unei luni sau a unui an calendaristic etc.



Definiție

Informații =

= fiecare dintre elementele noi, în raport cu cunoștințele prealabile, cuprinse în semnificația unui simbol sau a unui grup de simboluri: text scris, mesaj vorbit, imagini plastice, indicație a unui instrument etc. (cf. DEX).

Prin urmare, datele și informații nu sunt unul și același lucru:

- ◆ **datele** sunt informații primare – care au fost doar culese și înregistrate;
- ◆ **informațiile** sunt date prelucrate – care au fost în plus validate (corectate), organizate, sortate, relaționate.

Exemplu:

Să presupunem că trebuie să alcătuim o listă de referințe bibliografice privind elaborarea algoritmilor și evaluarea complexității lor. Pentru aceasta, vom începe prin a consulta fișierul bibliotecii școlii, al Bibliotecii Universitare și, eventual, vom căuta în una sau mai multe librării. De fiecare dată, vom nota rezultatele cercetărilor noastre. Tabelul 1 prezintă datele astfel culese. Tabelul 2 prezintă aceleași date dar corectate, ordonate alfabetic, corelate. Putem observa mai bine diferența dintre date (Tabelul 1) și informații (Tabelul 2).

Nr	Autor	Titlu	Editură	Localitate	An Ap
1	A. Aho, J.E Hopcroft, J.D. Ullman	The Design And Analysis Of Computer Alg.		Reading Mass.	1974
2	Johnsonbaugh Schaefer	Alg.	Pearson Prentice Hall	Upper Saddle Riv.	2004
3	Brainerd, Landweber	Theory Of Computation	John Wiley & Sons Inc.	New York	1974
4	Daniel I. A.	Cohen Intro. Comput.Th.	New York	John Wiley	1997
5	Cristian Sorin Calude	Theories Of Comp. Complexity	Elsevier	Amsterdam	1988
6	Th. Cormen, Etc.	Alg.	MIT Press,	Boston, Ma.	2000
7	C.M.-Vide, Victor Mitrana, Gh. Păun	Formal Languages And Applications	Springer-Verlag		2004

Tabelul 1: Date înregistrate

Nr	Autor	Titlu	Editură	Localitate	An Ap
1	A. AHO, J. E. HOPCROFT, J. D. ULLMAN	<i>The Design and Analysis of Computer Algorithms</i>	Addison-Wesley Publ. Co.	Reading Mass.	1974
2	W. S. BRAINERD, L H. LANDWEBER	<i>Theory Of Computation</i>	John Wiley & Sons Inc.	New York	1974
3	C. S. CALUDE	<i>Theories of Computational Complexity</i>	Elsevier Science Publ.	Amsterdam	1988

Tabelul 2: Date transformate în informații

Nr	Autor	Titlu	Editură	Localitate	An Ap
4	D. I. A. COHEN	<i>Introduction to Computation Theory</i> , 2 nd Ed.	John Wiley & Sons Inc.	New York	1997
5	T. H. CORMEN, C. E. LEISER- SON, R. L. RIVEST	<i>Algorithms</i>	MIT Press,	Boston, Ma.	2000
6	R.JOHNSONBAU GH M.SCHAEFER	<i>Algorithms</i>	Pearson Prentice Hall	Upper Sadle River, NJ.	2004
7	C. MARTIN- VIDE, V. MITRANA, GH. PĂUN (Eds.)	<i>Formal Languages and Applications</i>	Springer-Verlag Berlin	Heidelberg	2004

EXERCİTIU

Date versus informații

Identificați operațiile efectuate asupra datelor din Tabelul 1 pentru a le transforma în informațiile din Tabelul 2.

1.3. Modelarea datelor

1.3.1. Modelul de date

Dincolo de definițiile date până acum, ce este de fapt o bază de date? Este un obiect (asemenea numerelor, funcțiilor, mulțimilor)? Este o metodă (asemenea algoritmilor, procedurilor)?

O bază de date este în primul rând un **model** al microuniversului la care se referă.



Definiție

Model =

= (în sens strict) un sistem teoretic sau material cu ajutorul căruia pot fi studiate indirect proprietățile și transformările unui alt sistem, mai complex, cu care primul sistem prezintă o analogie;

= (în sens larg) ceea ce poate servi ca orientare pentru reproduceri (un tipar – cf. DEX).

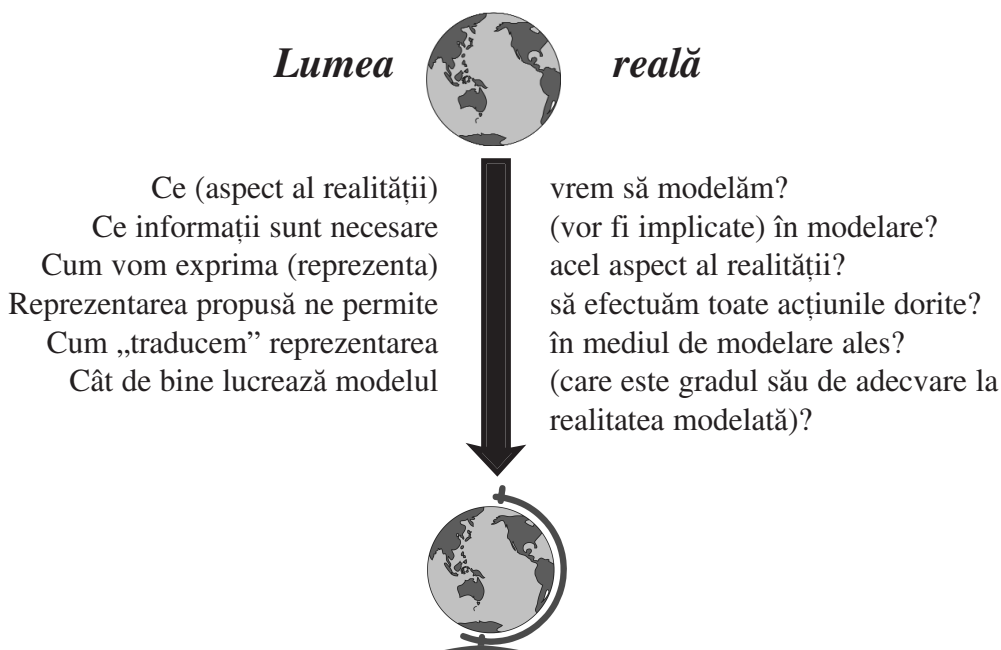


Figura 1: Procesul de construire a unui model

EXERCITII

1. Construirea unui model

- a) Comentați și ilustrați procesul de construire a unui model.
- b) Adaptați acest proces la cazul bazelor de date.

O bază de date înregistrează date despre un aspect al realității înconjurătoare, operând simplificările necesare și inerente: într-o bază de date privind angajații unei fabrici nu vom memora și informații despre romancierii sau soliștii preferați de fiecare angajat (deși și aceste informații pot avea relevanța lor), ci doar date despre starea civilă, studii, experiența anterioară etc. Ca urmare, un același aspect al realității înconjurătoare poate fi modelat în mai multe feluri, în funcție de scopul urmărit: activitatea dintr-o școală poate fi modelată din punctul de vedere al activității didactice, al evidenței școlare, al salarizării personalului etc. Cu alte cuvinte, alegerea unui anumit mod de reprezentare a situației reale pentru definirea bazei de date depinde de scopul pentru care trebuie construită baza de date.

2. Modele alternative

Un muzeu este o colecție de obiecte de artă (de diferite categorii: picturi, sculpturi, altele) expuse permanent sau prin expoziții tematice, itinerante etc. Unele obiecte pot fi scoase temporar din expoziție și depozitate în vederea restaurării. Muzeul dispune de personal administrativ dar și de personal specializat în organizarea de expoziții, restaurare, organizare de cursuri (de specialitate sau de inițiere).

Propuneți mai multe modele pentru activitățile unui muzeu.

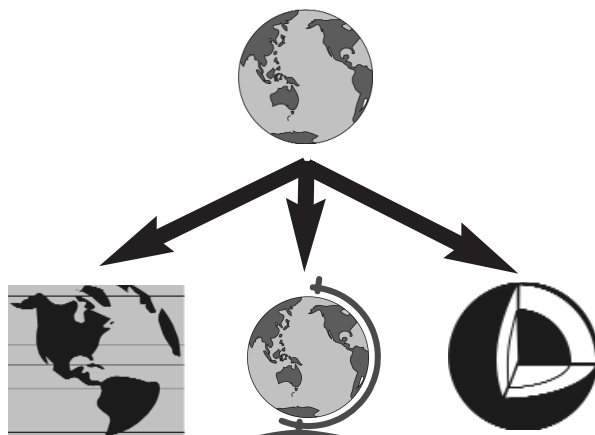


Figura 2: Modele multiple

3. Modele multiple

Alegeți un aspect din realitatea care vă este familiară și propuneți cel puțin trei modele ale acestuia; enunțați scopul corespunzător fiecărui model.

O bază de date oferă un anumit **grad de abstractizare a datelor** (asemenea celor mai multe limbaje de programare), ascunzând detaliile de implementare, detalii care nu sunt necesare celor mai mulți dintre utilizatori. Cu alte cuvinte, programele specifice unei baze de date nu depind de modul de stocare și accesare a datelor la nivel fizic. Acest concept se numește **independență a datelor**, se realizează cu ajutorul unui **model de date (Data Model*)** și este principalul mecanism care asigură partajarea datelor din baza de date între diferitele aplicații care le accesează.

* E.F. Codd este considerat a fi „părintele” conceptului de model de date, în general, și al conceptului de model de date relațional, în particular.



Definiție

Model =

= un ansamblu format din:

- 1) **o colecție de concepte** necesare pentru descrierea structurii bazei de date (a tipurilor de date incluse în baza de date, a relațiilor dintre ele și a restricțiilor – *Constraints* – pe care trebuie să le respecte);
- 2) **un set de operații de bază** (care să specifice modul de efectuare a extragerii și actualizării datelor din baza de date).

1.3.2. Modele de date: perspectivă istorică

Evoluția modelelor de date pentru bazele de date și SGBD-uri a fost sugestiv sintetizată de R.G.G. Canttelli în articolul său „What Are Next-Generation DB Systems?”, publicat în revista *Communications of the ACM*, în octombrie 1991: „Istoria informaticii a cunoscut multe generații de sisteme de gestiune a datelor, începând cu sistemele de fișiere indexate, continuând apoi cu sistemele de tip ierarhic și de tip rețea, iar – mai nou – cu sistemele relaționale. Acum suntem pe punctul de a intra într-o nouă generație de sisteme de gestiune a bazelor de date care oferă *administrare de obiecte* și care acceptă tipuri de date mult mai complexe”.

Cu toate că a generat o activitate de cercetare foarte susținută dar și o activitate practică, industrială extrem de productivă, domeniul bazelor de date este unul dintre cele mai tinere domenii ale informaticii. Este general acceptat faptul că „rădăcinile” sale trebuie căutate aproximativ acum 40 de ani (data apariției primului sistem comercial de gestiune a bazelor de date), în obiectivul fixat de președintele J.F. Kennedy pentru programul Apollo: aducerea primului om pe Lună până la sfârșitul anilor '60. În acel moment, nu exista niciun instrument informatic care să funcționeze efectiv și care să poată administra uriașele volume de date implicate în programul spațial. Ca urmare, North American Aviation (NAA), primul contractor al proiectului, a dezvoltat un software bazat pe o structură ierarhică (părțile se agregă în componente din ce în ce mai ample) denumit **GUAM (Generalized Update Access Method)**. Spre mijlocul anilor '60, IBM s-a alăturat NAA dezvoltând în continuare **GUAM** și producând unul dintre primele sisteme comerciale de gestiune a bazelor de date: **IMS (Information Management System)**. IBM a preluat modelul ierarhic pentru a respecta cerința de stocare a datelor pe benzi magnetice (deci în acces secvențial). Ulterior, această restricție a fost înlăturată și **IMS** continuă să fie principalul SGBD ierarhic utilizat de majoritatea calculatoarelor *mainframe*. Un calculator *mainframe*

este un calculator cu capacitate de memorie și viteză de lucru foarte mari, utilizat de marile corporații pentru a stoca volume foarte mari de date și pentru a coordona sute sau mii de terminale (inclusiv calculatoarele personale) conectate la el. Operarea unui *mainframe* necesită de obicei un personal specializat.

Construirea bazelor de date a cunoscut o evoluție foarte rapidă, trecând prin mai multe abordări, clasificate după cum urmează:

- ◆ sistemele de fișiere;
- ◆ sistemele prerelaționale (sau „istorice”, numite și navigante sau tradiționale – *legacy systems*): ierarhic și rețea;
- ◆ sistemul relațional;
- ◆ sistemele postrelaționale: orientat obiect și hibrid (obiect-relațional);
- ◆ sistemele semantice: multidimensional și logic (deductiv).

1.3.2.1. Sistemul de gestiune bazat pe fișiere (SGF)

Considerat de fapt un predecesor al sistemelor de gestiune a bazelor de date, SGF reprezenta o colecție de programe care realizau – fiecare – câte „un serviciu” pentru utilizatorii datelor (de obicei: generarea de rapoarte). Fiecare program își definea și își administra propriile date. Chiar dacă a avut numeroase dezavantaje (abordarea descentralizată în stocarea informațiilor, gradul mare de redundanță și dependență program-date), sistemul de gestiune bazat pe fișiere a constituit un salt semnificativ față de fișierele administrate manual: saltul de la abordarea informațională la cea informatică.

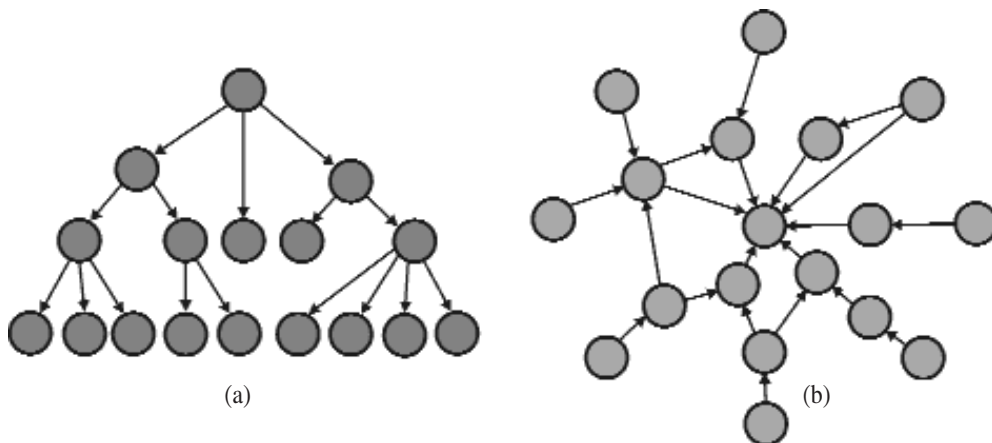


Figura 3: Modele de baze de date: (a) ierarhic; (b) rețea

1.3.2.2. Modelele prerelaționale

Aceste modele pot fi caracterizate ca modele de moment: au oferit soluții pentru problemele vremii lor, dar nu au avut un fundament teoretic puternic și riguros.

Atât în **modelul ierarhic**, cât și în **modelul rețea** datele erau reprezentate ca mulțimi de înregistrări (în sensul limbajului de programare *Pascal*: colecții de date de diferite tipuri: *Integer*, *Boolean*, *Real* etc.). Relațiile dintre ele erau reprezentate prin legături de tip pointer (adrese de locații fizice de memorie). Înregistrările care formau baza de date erau organizate:

- ◆ în modelul ierarhic: ca o mulțime de arbori;
- ◆ în modelul rețea: ca o mulțime de grafuri.

Ambele modele prerelaționale permiteau accesul la date de-a lungul unor drumuri (căi) predefinite, explicit stabilite la nivelul programelor de aplicații (de unde și numele de modele navigante). Ca urmare, orice modificare a structurii bazei de date antrena modificarea acestor căi în programele deja scrise. Exemple: pentru modelul ierarhic: **IMS** (amintit mai sus); pentru modelul rețea: **IDS II** (de la Honeywell), **IMAGE** (de la Hewlett Packard).

STUDIU DE CAZ

Modelarea activității didactice

Într-o facultate, *cadrele didactice* desfășoară *activități didactice* de *curs* sau *examen*; aceste activități sunt pentru *studenți* și se desfășoară în *locații* (amfiteatre sau laboratoare). De asemenea, cadrele didactice participă la *proiecte de cercetare științifică*. Figura 4 prezintă modelul ierarhic al facultății; Figura 5 prezintă modelul rețea.

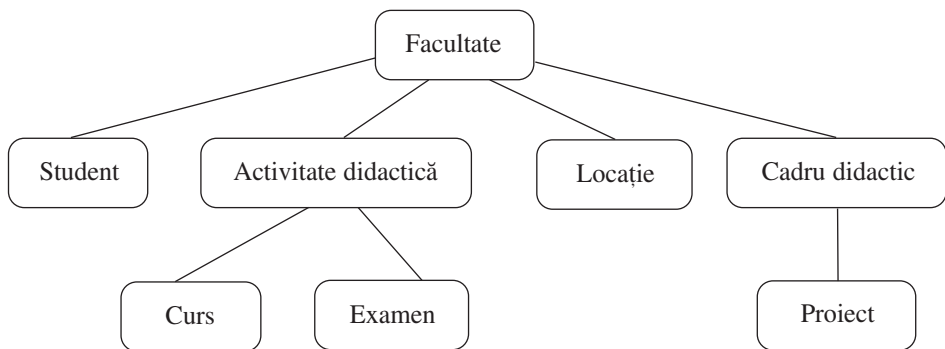


Figura 4: Modelul ierarhic

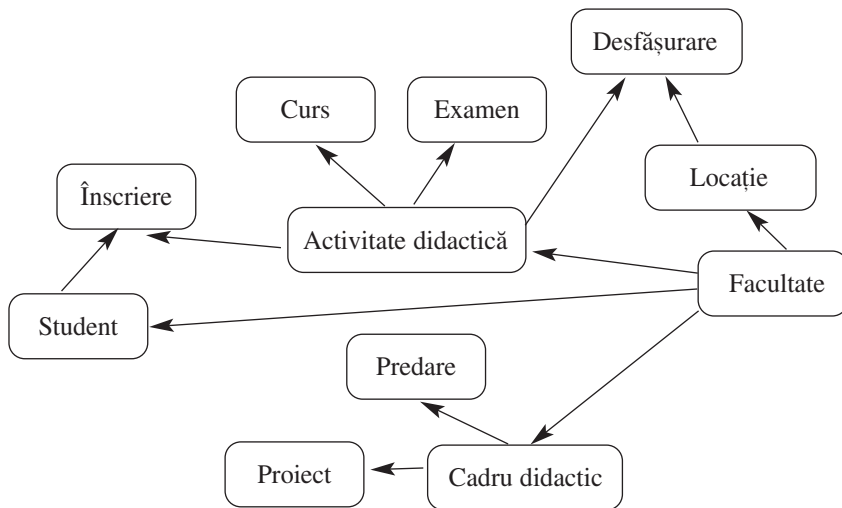
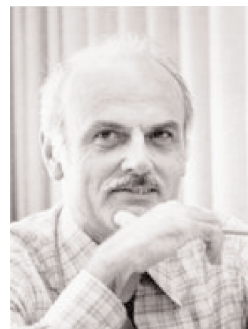


Figura 5: Modelul rețea

1.3.2.3. Modelul relațional

Considerat drept cel mai important eveniment din istoria bazelor de date, apariția modelului relațional s-a produs în iunie 1970, odată cu publicarea – în revista *Communications of the ACM* – a articolului fundamental al lui Edgar Frank Codd* (de la IBM Research Laboratory): „A Relational Model of Data for Large Shared Databanks”. În acest articol, autorul aplica o serie de concepte din algebra relațională pentru a rezolva problemele legate de stocarea volumelor mari de date și enunța „celebrele” 12 reguli (condiții) pe care trebuie să le îndeplinească un SGBD pentru a fi declarat relațional.



Să amintim însă existența unui precursor: modelul bazat pe teoria mulțimilor, propus de D.L. Childs în articolul său: „Feasibility of a Set-Theoretical Data Structure”, apărut în 1968.

Cele mai importante prototipuri de sisteme de gestiune a bazelor de date de tip relațional au fost:

- ◆ **System R**, dezvoltat la San Jose Research Laboratory din California spre sfârșitul anilor '70. Acest model a condus la:

* E.F. Codd (foto) s-a născut la 23 august 1923 în Portland, Marea Britanie, și a murit în 18 aprilie 2003, în Florida. A făcut studii de matematică și chimie la Oxford și s-a mutat în Statele Unite, în 1948, pentru a lucra la IBM. A introdus termenul **OLAP (OnLine Analytical Processing)** și a impus modelul relațional; a avut, de asemenea, contribuții în domeniul modelelor de calculabilitate prin lucrările sale privind automatele celulare. A obținut de două ori Premiul Turing: în 1981 și 1994.

- apariția unui limbaj structurat de interogare a bazelor de date: **SQL**.
- producerea mai multor SGBD-uri relaționale comerciale: **DB2** și **SQL/DS** de la IBM și, respectiv, **ORACLE** de la Oracle Corporation (în deceniul 9 al secolului trecut).

◆ **INGRES (Interactive Graphics Retrival System)**, dezvoltat la Universitatea Berkeley din California.

◆ **Peterlee Relational Test Vehicle**, dezvoltat la IBM UK Centre din Peterlee, Marea Britanie.

Numărul sistemelor relaționale comerciale a ajuns acum la câteva sute, dintre care cele mai cunoscute sunt: **DB2** (de la IBM), **Ingres II** (de la Computer Associates International Inc.), **Oracle 10g** (de la Oracle Corporation), **Ms Access**, **FoxPro** (de la Microsoft), **Paradox**, **Visual dBase** (de la Borland), **Sybase Adapted Server** (de la Sybase Inc.). Succesul acestui model continuă să fie atât de mare încât multe sisteme nerelaționale oferă acum și o interfață cu utilizatorii de tip relațional, indiferent de modelul de date pe care se bazează de fapt.

Modelul relațional s-a dovedit a fi și un instrument didactic ideal de prezentare a principiilor bazelor de date, tocmai datorită fundamentării sale riguroase pe principii logice și matematice.

Ce este de fapt un model relațional de date? Informal, îl putem defini ca pe un model în care:

- ◆ datele sunt percepute de utilizatori ca niște tabele și numai ca niște tabele;
- ◆ operațiile disponibile pentru utilizatori (spre exemplu, pentru obținerea informațiilor) sunt operații care generează noi tabele pe baza tabelor vechi: operația de selecție (SELECT) extrage o submulțime de rânduri dintr-o tabelă, operația de proiecție (PROJECT) extrage o submulțime de coloane, operația de juxtapunere (JOIN) asociază două tabele pe baza valorilor identice pe care le conțin în anumite coloane, de asemenea identice; or, toate aceste submulțimi rezultate pot fi privite și ele însele ca niște tabele.

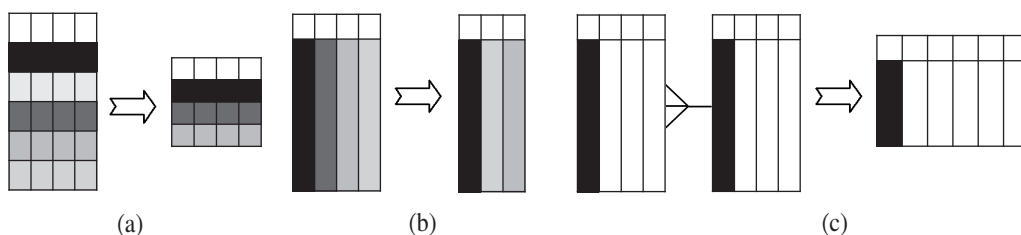


Figura 6: Operații cu tabele: (a) selecție; (b) proiecție, (c) asociere

Numele modelului (model **relațional**) provine de la conceptul matematic de **relație**. Așa cum o funcție $f : \{1, 2, \dots, n\} \subset \mathbb{N} \rightarrow \mathbb{R}$ are mai multe reprezentări convenționale, dintre care cea mai comodă este cea de vector, tot astfel relația poate avea mai multe reprezentări, una dintre ele fiind tabela. **Din acest motiv, cel puțin la nivel informal, termenii de relație și tabelă pot fi considerați sinonimi.**

Principalele concepte cu care lucrează modelul de date de tip relațional sunt (exemplificarea se face pentru baza de date asociată unui liceu):

- ◆ **entitatea** (*Profesori, Elevi, PersonalAuxiliar, Clase* etc.),
- ◆ **relația dintre entități** (*PredăLa, AreDirigintePe, AreLocIn* etc.),
- ◆ **atributul** (*Nume, GradDidactic, DataNașterii, NrLocuri, Locație* etc.).

EXERCIȚIU

Modelele ierarhic, rețea și relațional

- a) Realizați un tabel comparativ al modelelor de date ierarhic, rețea și relațional, urmărind organizarea datelor și a legăturilor dintre ele.
- b) Enumerați cel puțin trei avantaje și trei dezavantaje ale modelelor ierarhic, respectiv rețea, față de modelul relațional.

1.3.2.4. Modelele postrelaționale

Chiar dacă se regăsește în descrierea unor situații reale, cu organizare intrinsec piramidală, modelul ierarhic și-a atins rapid limitele. La fel, modelul relațional a devenit impropriu pentru rezolvarea unor probleme din realitatea înconjurătoare care presupun manipularea unor volume uriașe de informație, a unei mari varietăți de tipuri de date: hărți meteorologice sau geografice necesare previziunilor meteorologice sau dirijării traficului, imagini transmise prin satelit utilizate în măsurarea factorilor poluanți, date neconvenționale pentru proiectarea asistată de calculator în inginerie sau arhitectură, serii dinamice implicate în tranzacțiile bursiere sau bancare, stocarea obiectelor binare mari (*BLOBs = Binary Large Objects*) necesare în digitalizarea informației conținută în fișierele audio sau video. Au apărut astfel și s-au dezvoltat modelele postrelaționale, de generația a treia: **modelul orientat obiect** și **modelul obiect-relațional**.

(I) **Modelul orientat obiect** permite înglobarea semanticii obiectelor celor mai variate, la fel ca în limbajele de programare orientate-obiect. De altfel, una dintre deosebirile majore față de modelul relațional constă în distanțarea de conceptul de **independență** față de limbajele de programare și dezvoltarea conceptului de **integrare** a limbajelor de programare în sistemul de gestiune a bazei de date (invocarea unor funcții C++ mai degrabă decât înglobarea unui limbaj special pentru interogarea datelor, ca de exemplu SQL). Acest fapt a fost determinat de:

- ◆ utilizarea aproape exclusivă a limbajelor de programare orientate-obiect pentru dezvoltarea aplicațiilor software;
- ◆ includerea în aproape orice aplicație software a unei baze de date ca element fundamental al acesteia.

Cele mai cunoscute prototipuri de baze de date orientate-obiect sunt: **OPENOODB** (de la Texas Instruments), **IRIS** (de la Hewlett Packard), iar ca variantă comercială: **GEMSTONE/OPAL** (de la GemStone Systems), **VERSANT** (de la Versant Object Technology). Deși cu o cotă de piață semnificativ inferioară sistemului relațional (150 milioane dolari față de 10 miliarde, numai în SUA în anul 1999), modelul orientat-obiect este creditat cu o creștere anuală extrem de rapidă: 50%.

În ciuda caracterului intuitiv și a altor avantaje evidente ale modelului orientat-obiect, modelul relațional continuă să domine piața sistemelor de gestiune a bazelor de date. Motivele sunt numeroase: fundamentarea matematică riguroasă, simplitatea, volumul mare de date deja stocate după acest model și costul enorm al migrării spre un model complet diferit.

(II) Modelul obiect-relațional extinde modelul relațional, oferind un set de tipuri de date mai bogat, și include și orientarea obiect. Se încearcă astfel combinarea avantajelor celor două abordări, cea relațională și cea orientată-obiect: atributele și instanțele entităților pot avea tipuri complexe și pot evita unele dintre restricțiile specifice modelului relațional. De exemplu, în timp ce în modelul relațional fiecare atribut trebuie să ia pentru fiecare instanță a unei entități o valoare și numai una din domeniul lui de definiție, în acest model poate lua un subset de valori (de exemplu: pentru un angajat oarecare, atributul *Telefon* poate lua ca valori numărul telefonului fix de acasă și de la serviciu, al telefonului mobil propriu și de serviciu, dacă angajatul dispune de toate patru).

Cel mai cunoscut exemplu: **Informix Universal Server** care combină tehnologiile relaționale și orientate obiect din două produse preexistente: **Informix** și **Illustra**.

Principalele avantaje și dezavantaje ale modelelor de date (și ale sistemelor de gestiune a bazelor de date corespunzătoare) reies din Figura 7 (vezi [18]): modelul relațional permite realizarea – chiar simultană – a unor interogări variate și rapide, dar complexitatea datelor stocate nu diferă prea mult de complexitatea datelor memorate în baze de date de tip ierarhic sau rețea; cu modelul orientat obiect se poate stoca informație variată și complexă (de la texte la sunete și imagini), dar viteza de interogare (în cazul imaginilor și mai ales al sunetelor) este foarte scăzută; modelul care pare să elimine toate dezavantajele și să cumuleze toate avantajele modelelor anterioare este modelul obiect-relațional.

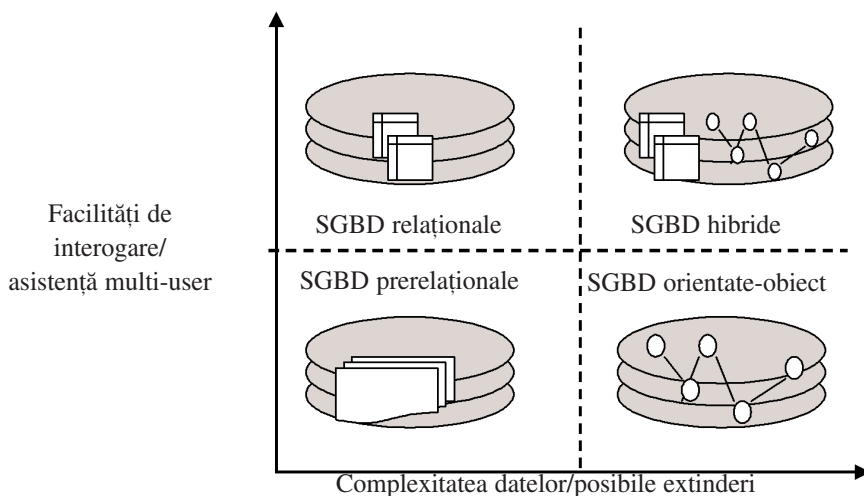


Figura 7: Clasificarea Stonebraker pentru sistemele de gestiune a bazelor de date

TEST: MODELE DE DATE

Citiți întrebările de mai jos și alegeți varianta corectă de răspuns:

1. Un model de date este:
 - a) mecanismul care asigură partajarea datelor dintr-o bază de date între aplicațiile care o accesează;
 - b) tehnica prin care pot fi organizate informațiile dintr-o întreprindere.
2. Modelul ierarhic și modelul rețea sunt:
 - a) modele relaționale;
 - b) modele prerelaționale;
 - c) modele hibride.
3. Asociați fiecare tip de model de date cu conceptul teoretic pe care se bazează:

	Model		Concept
1.	Ierarhic	A.	Relație
2.	Rețea	B.	Arbore
3.	Relațional	C.	Graf