

Proiectarea bazelor de date distribuite

Lect. Cristina IONITA, prep. Petrisor OPREA
Catedra de Informatica Economica, A.S.E. Bucuresti

*Ca urmare a cerintelor de crestere a disponibilitatii, sigurantei si flexibilitatii sistemelor de baze de date (BD) precum si a progreselor înregistrate în tehnologia BD si în domeniul comunicatiilor, sistemele de baze de date distribuite (BDD) sunt în prezent larg raspândite. Dezvoltarea BDD este motivata atât tehnologic cât si organizatoric întrucât elimina o parte din neajunsurile bazelor de date centralizate si sunt bine venite pentru descentralizarea structurii organizatorice în cadrul unei unitati economice. Cercetarile în domeniul BDD au început în anii '70 dar adevaratele sisteme distribuite au început sa apara pe piata pe la mijlocul anilor '80: ORACLE SQL*STAR, INGRES/STAR, VAX Data Distribution etc. În prezent, majoritatea SGBD-urilor au în componenta module pentru asigurarea distributiei datelor. Proiectarea unei BDD necesita metodologii si tehnici noi care sa tina seama de obiectivele specifice BDD. Proiectarea BDD este dificil de formalizat datorita complexitatii problemei de distributie a datelor si a restrictiilor de realizare.*

Cuvinte cheie: baze de date, model distribuit, metode de proiectare.

1. Introducere

Pentru a asigura respectarea obiectivelor specifice ale BDD si anume: cresterea sigurantei sistemului si a disponibilitatii datelor, descentralizarea resurselor sistemului, o mai buna utilizare a acestora precum si sporirea adaptabilitatii sistemului la modificarile din structura organizatorica, în proiectarea BDD se urmareste asigurarea respectarii urmatoarelor **principii**:

↳ *Maximizarea prelucrarii locale a datelor*, care presupune plasarea datelor cât mai aproape de aplicatiile care le solicita. Se apreciaza ca într-o BDD corect proiectata aproximativ 90% din volumul de date trebuie sa fie accesate local si numai 10% sa fie accesate de la distanta.

↳ *Asigurarea unui nivel crescut de siguranta si disponibilitate a datelor*, care poate fi realizat prin replicarea datelor pe mai multe statii. În aceste conditii, sistemul poate utiliza o copie alternativa atunci când cea care trebuia sa fie accesata în conditii normale nu este disponibila.

↳ *Procesarea paralela a datelor*. Distribuirea bazelor de date ofera posibilitatea de a utiliza eficient capacitatile procesoarelor din fiecare statie pentru a maximiza gradul de paralelism în executia aplicatiilor. Cresterea numarului de prelucrari realizate în paralel intra însa în contradictie cu principiul de maximizare a prelucrarilor locale. În proiectarea BDD trebuie sa asigure un raport optim între cele doua tipuri de prelucrari.

Tipuri de proiectare. Proiectarea BDD poate fi realizata, ca si în cazul BD centralizate, în maniera top-down si/sau bottom-up:

- Proiectarea top-down urmareste asigurarea unei distribuii optime a datelor si este utilizata atunci când proiectarea BDD începe de la zero. În general în urma abordarii top-down sistemele rezultate sunt omogene.

- Proiectarea bottom-up a BDD este recomandata în cazul în care BD locale exista si trebuie integrate într-un sistem unitar. În acest caz obiectivul principal urmarit în proiectarea BDD îl constituie asigurarea unei cooperari cât mai bune între BD existente care sunt de regula eterogene.

- Proiectarea mixta, partial top-down si respectiv partial bottom-up este reco-mandata în majoritatea cazurilor pentru ca raspunde mai bine situatiilor practice.

2. Proiectarea top-down

Proiectarea descendenta, reprezentata schematic în *figura 1*, este un proces de rafinare succesiva a specificatiilor globale ale BDD, independente de sistem, în specificatii detaliate care sa tina cont de restrictiile impuse de sistem. Punctul de plecare îl constituie identificarea si analiza cerintelor informati-onale ale sistemului care va exploata BDD. Pornind de la aceste cerinte se va proiecta mai întâi schema globala a BDD independent de restrictiile fizice de distribuire si implementare. Urmeaza proiectarea distribu-irii datelor, care va pune în evidenta modul de fragmentare a BDD si de alocare a fragmentelor la statii. Abordarea descen-denta se va încheia cu proiectarea BD locale pentru fiecare statie în parte.

Tehnicile de proiectare a schemei globale a BDD si a BD locale sunt asemanatoare cu cele utilizate în cazul BD centralizate. Spe-cifice BDD sunt aspectele legate de proiect-tarea schemei de fragmentare si a schemei de alocare a fragmentelor la statii.

a. Proiectarea fragmentarii. Descompu-nera schemei globale în frag-mente poate fi realizata aplicând doua tipuri principale de fragmentare: fragmentarea orizontala si frag-mentarea verticala.

Fragmentarea orizontala consta în partiti-onarea tuplurilor unei colectii globale în subseturi, pe baza unor predicate de selectie. Predicatul de selectie denumit si *calificare* este asociat fiecarui fragment si defineste proprietatea care a stat la baza gruparii tu-plurilor în cadrul fragmentului. Ratiunea fragmentarii orizontale este de a produce fragmente cu potential maxim de procesare locala, fiecare fragment obtinut fiind alocat la

statia unde este cel mai utilizat. Alegerea predicatelor de selectie nu este atât de simpla cum ar putea sa para la prima vedere. Trebuie luate în considerare atât de propri-etatile logice ale datelor cât si de carac-teristicile aplicatiilor care le exploateaza (cum ar fi numarul de accese de regasire, respectiv de actualizare a seturilor de date). Corelarea aspectelor logice ale datelor cu ca-racteristicile aplicatiilor este adesea dificil de realizat.

Identificarea setului de predicate este arbi-trara si se realizeaza tinând cont de cerintele informationale ale sistemului.

Pentru determinarea unui set de predicate de fragmentare Pr' , care sa realizeze fragmen-tarea corecta si eficienta a unei colectii de date globale R pornind de la un set de predi-cate, Pr , determinat în mod arbitrar, se poate aplica un algoritm constituit din doi pasi:

P1. Initializarea. Se cauta un predicat $p_i \in Pr$ care partitioneaza tuplurile colectiei R în doua fragmente referite diferit de cel puțin o aplicatie. Predicatul p_i va fi adaugat la multimea Pr' si va fi eliminat din multimea Pr : $Pr' = \{p_i\}$, $Pr = Pr - p_i$

P2. Adaugarea iterativa a predicatelor simple la multimea Pr' pâna când devine completa. Se cauta un predicat $p_j \in Pr$ care partitioneaza o parte din fragmentele definite în acord cu predicatele mintern construite pe baza setului Pr' , în doua parti referite diferit de cel puțin o aplicatie. Predicatul p_j va fi adaugat la multimea Pr' si va fi eliminat din multimea Pr :

$$Pr' = Pr' \cup p_j, Pr = Pr - p_j$$

Determinarea unui set complet si minimal de predicate poate fi destul de costisitoare în anumite cazuri. Pentru a evita aceste situatii este recomandat ca în stabilirea setului de predicate simple sa fie luate în considerare numai aplicatiile importante, care acceseaza frecvent o parte din tuplurile unei relatii globale si sa nu separam fragmente care au caracteristici asemanatoare.

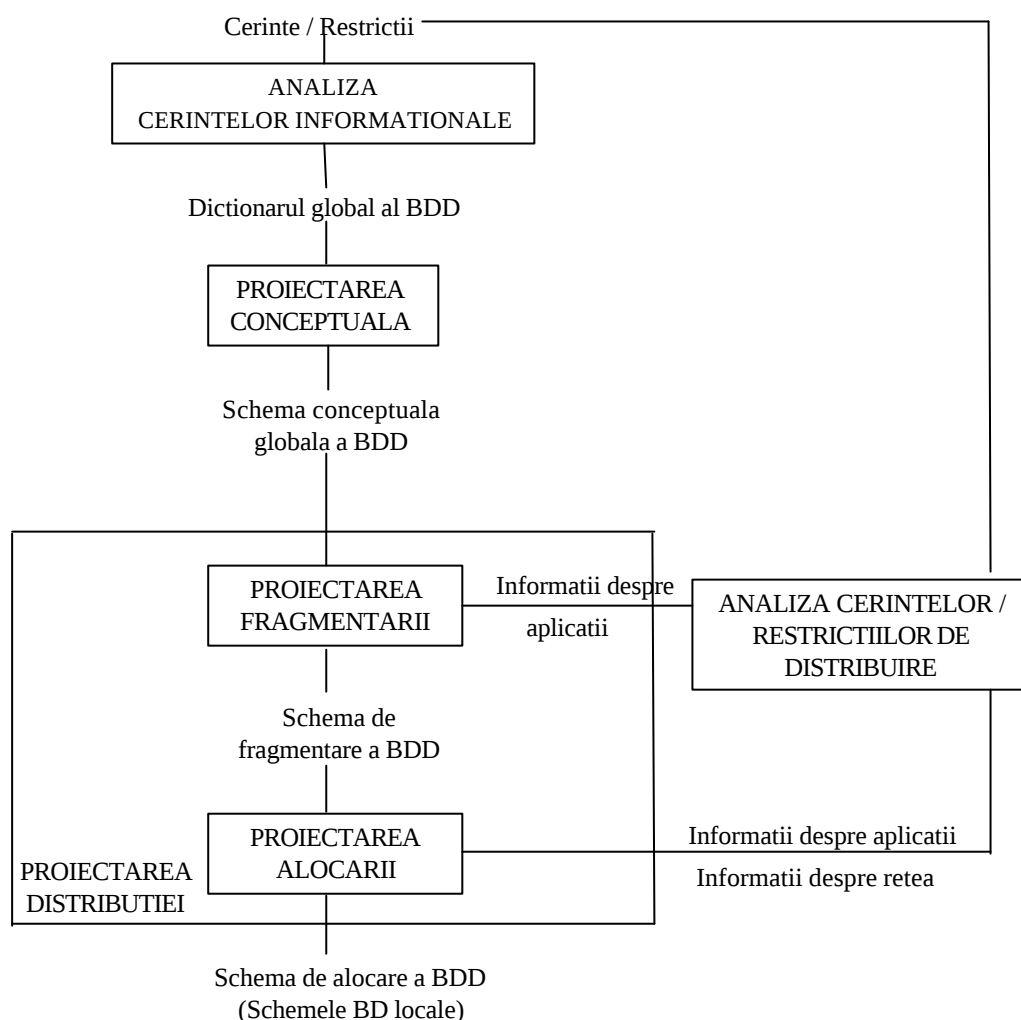


Fig. 1. Proiectarea top-down a BDD

Fragmentarea verticala se realizeaza prin împartirea atributelor unei relatii globale în seturi astfel încât mai multe aplicatii sa poa-ta fi executate utilizând un singur fragment.

Ratiunea pentru fragmentarea verticala este aceea de a grupa împreuna attributele frecvent utilizate împreuna. O buna fragmentare verticala se realizeaza când fiecare aplicatie utilizeaza un singur fragment, altfel aplicatiile care necesita accesarea mai multor fragmente verticale vor

avea performante mai scazute, deoarece trebuie sa realizeze una sau mai multe jonctiuni si, implicit, nu-meroase accese la distanta. În aceste conditii, trebuie asigurat un echilibru între beneficiul potential datorat posibilitatii de plasare a fiecarui fragment în apropierea aplicatiilor care îl utilizeaza cel mai frecvent si pierderile potentiale datorate faptului ca exista aplicatii care utilizeaza doua sau mai multe fragmente aflate pe statii diferite.

Fragmentarea verticala este considerata mai dificila decât cea orizontala pentru ca exista mai multe alternative. Posibilitatile de grupare a atributelor cresc combinatorial cu numarul de attribute al tabelii de date. În schimb conditia de corectitudine a fragmentarii este mai usor de realizat si de verificat, fara a apela la algoritmi speciali.

Fragmentarea mixta se realizeaza prin aplicarea succesiva a operatiilor de fragmentare orizontala si verticala. Astfel, fragmentele mixte pot fi obtinute prin aplicarea fragmentarii orizontale la un fragment vertical, respectiv prin aplicarea fragmentarii verticale la un fragment orizontal. Aceste operatii pot fi repetate recursiv generând arbori de fragmentare de orice complexitate. În practica este recomandat sa existe cel mult doua niveluri de fragmentare.

b. Proiectarea alocarii. Alocarea fragmentelor pe statii este stâns legata de replicarea datelor din BDD. La alocare proiectantul trebuie sa decida daca fragmentele BDD vor fi replicate si gradul de replicare a acestora. Replicarea are ca efect pozitiv cresterea sigurantei sistemului si regasirea mai rapida a datelor. În conditiile replicarii datelor, mentinerea consistentei datelor în BD este mai dificil de asigurat pentru ca actualizarea unui fragment trebuie sa fie propagata la toate copiile fragmentului. Decizia replicarii unui fragment este influentata, în mare masura, de raportul dintre numarul acceselor de regasire si cel al acceselor de actualizare pentru fragmentul respectiv.

Alocarea fragmentelor pe statiile retelei, în functie de metodele utilizate, poate avea ca rezultat o alocare finala neredundanta sau redundanta. *Alocarea finala neredundanta* este recomandata când frecventa de actualizare a datelor este mai mare decât frecventa de regasire a datelor. *Alocarea finala redundanta* este recomandata când frecventa de regasire a datelor este mai mare decât frecventa de actualizare a datelor. Replicarea datelor este

mai convenabila în cazul sistemelor care admit inconsistente temporare ale datelor.

Problema de alocare

Distribuirea eficienta a fragmentelor presupune asigurarea unui echilibru între *costuri* (de stocare, prelucrare si transmitere a datelor), *performante* (în special timpul de raspuns) si *restrictii de distribuire* a datelor. Realizarea alocarii fragmentelor BDD este o problema de optimizare complexa ce presupune distribuirea unei multimi de n fragmente $F = \{F_1, F_2, \dots, F_n\}$ exploatate de o multime de k aplicatii $A = \{A_1, A_2, \dots, A_k\}$ pe o multime de m statii $S = \{S_1, S_2, \dots, S_m\}$, luând în calcul urmatoarele categorii de informatii (variabile) [OZSU94]:

1. Informatii despre fragmentele BDD, cum ar fi: dimensiunea fragmentelor, selectivitatea fragmentelor;
2. Informatii despre cererile (aplicatiile) care vor exploata fragmentele, cum ar fi: numarul acceselor de citire/actualizare ale fiecărei cereri pentru fiecare fragment, timpul maxim acceptabil de raspuns;
3. Informatii privind statiile retelei, cum ar fi: unitatea de cost pentru stocarea/prelucrarea datelor pe o statie, capacitatea maxima de stocare/prelucrare a datelor pe o statie;
4. Informatii privind comunicarea în retea, cum ar fi: costul comunicarii între statii, dimensiunea frame-urilor, overhead-ul de comunicatie.

În general, costul alocarii unui fragment R_i la o statie j poate fi determinat astfel:

$$C_{ij} = s_{ij} + \sum_{k=1}^p l_{kij} + \sum_{k=1}^p \sum_{\substack{j'=1 \\ j' \neq j}}^n a_{kij'}, \quad i=1,m; j=1,n$$

unde:

- s_{ij} – costul stocarii fragmentului R_i la statia j ;
- l_{ijk} – costul acceselor locale la fragmentul R_i de pe statia j a aplicatiei k ;
- a_{ij} – costul acceselor de actualizare a fragmentului R_i la statia j realizate de aplicatiile k de pe celelalte statii ($j \neq j'$);
- p – numarul de aplicatii; m – numarul de fragmente; n – numarul de statii.

Replicarea unui fragment R_i la statia j se justifica daca costul alocarii fragmentului este

mai mic decât costul accesului la distanță a aplicațiilor k de la stația j , la celelalte copii ale fragmentului aflate pe stațiile j' ($j' \neq j$):

$$C_{ij} < \sum_k d_{kij}$$

În literatura de specialitate sunt prezentate metode de alocare a fragmentelor BDD care iau în calcul, integral sau parțial, categoriile de informații prezentate. Cele mai cunoscute sunt metodele propuse de Stefano Ceri și Giuseppe Pelagatti în [CERI85] pentru evaluarea costurilor și respectiv beneficiilor de alocare a fragmentelor și anume:

- *Metoda cele mai bune alegeri* constă în comensurarea beneficiului fiecărei alocări posibile a unui fragment și alegerea stației cu cea mai bună măsură. Metoda realizează o alocare finală neredundantă.
- *Metoda selectării stațiilor profitabile* presupune determinarea unui set de stații pentru care beneficiul alocării unei copii a fragmentului este mai mare decât costul alocării
- *Metoda replicării progresive* presupune construirea inițială a unei soluții neredundante și apoi introducerea progresivă a copiilor replicate, începând cu cea mai profitabilă. Replicarea progresivă se va încheia atunci când nici o replicare posibilă nu mai este profitabilă. Metodele sunt empirice, punând accent pe evidențierea criteriilor de alocare a fragmentelor și nu pe măsurarea exactă a costurilor și beneficiilor, acestea fiind exprimate simplificat numai prin frecvențele de acces la fragmente.

3. Proiectarea bottom-up

În cadrul proiectării bottom-up, reprezentată schematic în figura 2, punctul de plecare îl constituie schemele BD locale. Se presupune ca specificațiile BD pentru fiecare stație există, fie pentru ca BD locale există deja și trebuie interconectate într-un sistem de multi-BD, fie pentru ca specificațiile conceptuale ale BD au fost făcute pentru fiecare stație, în mod independent, și trebuie integrate într-o schemă

globală.

Prin integrarea schemelor BD existente se urmărește corelarea descrierilor comune de date și rezolvarea conflictelor care pot să apară ca urmare a diferențelor. Atunci când BD existente sunt integrate într-o BDD este posibil ca acestea să utilizeze SGBD-uri diferite. Un sistem eterogen adăuga, deci, la complexitatea integrării datelor și necesitatea translatarii lor între diferite reprezentări. În acest caz, abordarea ascendentă a BDD presupune parcurgerea următorilor pași: alegerea unui model comun pentru a descrie schema globală; translatarea fiecărei scheme locale în modelul de date ales; integrarea schemelor locale într-o schemă globală.

Unul dintre cele mai utilizate modele pentru descrierea schemei globale îl constituie modelul entitate-asociere extins cu asocieri de tip generalizare și specializare. Generalizarea ne ajută să punem în evidență părțile comune ale entităților iar specializarea diferențele între entități. De exemplu, în figura 3a este reprezentată entitatea E în viziunea a două scheme locale: $E(A_1, A_2, A_3, A_4, A_5)$ și $E(A_1, A_2, A_3, A_6, A_7)$, iar în figura 3b este prezentat arborele de generalizare. Integrarea unui set de BD locale eterogene constituie o problemă complexă, dificil de formalizat pentru că situațiile de la care se pornesc sunt foarte variate. În literatura de specialitate majoritatea tehnicilor de proiectare se referă la abordarea top-down și numai un număr redus tratează proiectarea bottom-up, deși în practică aceasta abordare este cea mai frecventă. Un set de activități pe care se bazează integrarea schemelor BD locale a fost identificat în [CERI92], și anume:

- *Identificarea similarităților.* Părțile comune ale schemelor locale pot fi identificate relativ ușor datorită similarității numelor de entități și atribute, a descrierii structurii sau chiar prin identificarea unui set de valori comune.
- *Identificarea diferențelor și a conflictelor posibile.* Principalele categorii de diferențe

care pot genera conflicte se refera la diferite de nume (sinonime respectiv omonime), diferite de descriere a datelor (natura, dimensiune), diferite de domeniu de valori,

diferente structurale. Multe din aceste conflicte pot fi rezolvate înainte de integrarea schemelor locale prin modificarea corespunzatoare a acestor scheme

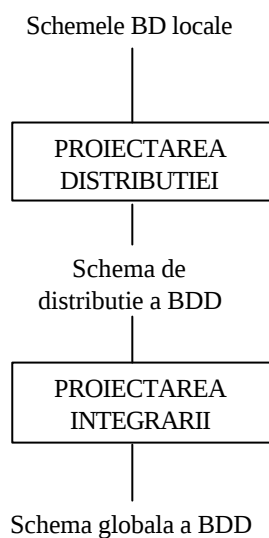
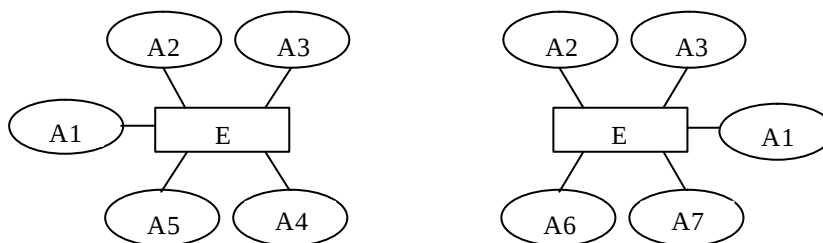
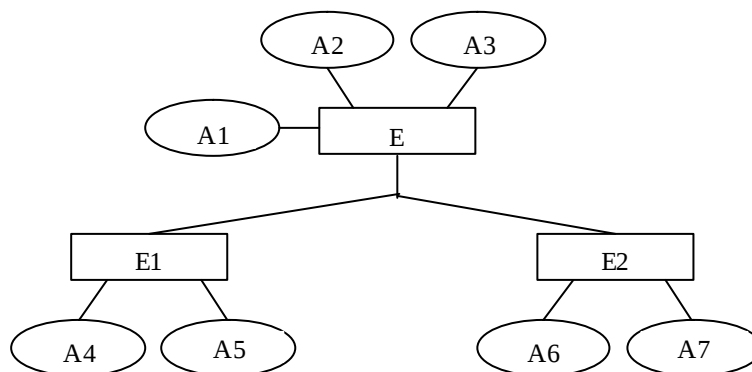


Fig. 2. Proiectarea bottom-up a BDD



a. Viziunea schemelor locale



b. Arborele de generalizare

Fig. 3. Integrarea schemelor locale

O parte însă necesită includerea în schema globală a unor informații suplimentare pentru tratarea conflictelor (tabele de corespondență, formule de conversie, comportamentul diferit al

▪ *Determinarea politicilor de soluționare a inconsistentelor.* Inconsistențele determinate de replicarea datelor sunt frecvent întâlnite într-un sistem de multi-BD și sunt datorate deficiențelor de integrare a schemelor locale, nesincronizării operațiilor de actualizare sau erorilor de refacere a BD după incidente. Proiectantul BDD va trebui să decidă politica de tratare a situațiilor în care se înregistrează valori diferite ale unor obiecte aparent identice aflate pe stații diferite: utilizarea uneia din valorile inconsistente, fără să avertizeze utilizatorul; prezentarea tuturor valorilor inconsistente utilizatorului, indicând și sursa informațiilor; utilizarea celei mai recente valori; utilizarea valorii din sistemul cel mai de „încredere”; utilizarea unor mărimi agregate determinate pe baza valorilor inconsistente (medie aritmetică, maxim, minim etc.).

Concluzii

Sistemele de BDD prezintă o serie de avantaje legate în special de creșterea performanțelor sistemului, de reducerea relativă a costurilor prelucrării și transmiterii a datelor și nu în ultimul rând de faptul că un sistem de BDD se potrivește mai bine structurii organizatorice dintr-o unitate economică. Dezavantajele sunt legate de complexitatea ridicată de proiectare a

aplicațiilor). Dacă SGBD-ul distribuit nu oferă facilități pentru tratarea conflictelor, acestea vor trebui gestionate de aplicații, lucru mai dificil de realizat.

unui astfel de sistem care suprapune peste complexitatea unei rețele de calculatoare problemele legate de prelucrarea distribuită a datelor și asigurarea unui nivel înalt de transparență a distribuției.

Procesul de realizare a unui sistem de BDD este complex și se justifică în general în una din următoarele situații: atunci când se realizează BD de dimensiuni mari, care oferă suport pentru majoritatea activităților desfășurate într-o organizație, în situația în care se preconizează extinderi frecvente ale sistemului existent sau pentru interconectarea unui set de BD existente.

Bibliografie

1. S. Ceri, G. Pelagatti, *Distributed Databases, Principles and Systems*, International Student Edition, McGraw Hill, 1985
2. S. Ceri, B. Pernici, G. Wiederhold, *Distributed Database Design Methodologies*, Proceedings of The Second International Conference on Systems integration, New Jersey, April 23-26, 1992
3. I. Lungu, C. Bodea, G. Badescu, C. Ionita, *Baye de date. Organizare, proiectare și implementare*, Ed. ALL, 1995
4. M. Tamer Özsu, P. Valduriez, *Distributed Database Management*, Prentice Hall, 1994