

Modele de date utilizate în bazele de date pentru prelucrări grafice

Sef lucrări dr. ing. Marius Dorian ZAHARIA, Catedra de Calculatoare Universitatea POLITEHNICA Bucuresti

Lucrarea prezintă principalele modele de date specifice sistemelor de gestiune a bazelor de date referindu-se în special la modelele adecvate bazelor de date spațiale și la modelul orientat spre obiecte.

Cuvinte cheie: *model de date, baze de date, baze de date spațiale.*

1. Introducere

Bazele de date utilizate în prelucrări grafice își găsesc aplicații în domenii diverse cum ar fi:

Proiectarea și fabricația asistate de calculator precum și în alte aplicații de automatizare a proiectării (CASE). Utilizatorii pot avea acces (eventual concurrent) la modele de obiecte 3D (în cazul proiectării mecanice) sau 2½ D (în cazul proiectării asistate de calculator în domeniul electronicii). Aceste categorii de aplicații trebuie să permită descrierea structurală a unui proiect, simularea funcționalității lui, sinteza și gestiunea versiunilor proiectului.

Aplicațiile de birotică, publicare asistată de calculator (CAP) realizează gestiunea documentelor din diferite domenii de activitate, stocarea, manipularea și regăsirea documentelor complexe, inclusiv a celor cu "comportare dinamică".

În domeniul graficii pe calculator există categorii numeroase de aplicații front-end (client) destinate furnizării unor interfețe grafice pentru diferite sisteme de gestiune a bazelor de date (back-end servers). Sistemele de vizualizare științifică constituie un caz important de astfel de aplicații.

Din categoria bazelor de date pentru prelucrări grafice fac parte și sistemele de gestiune a informației geografice (GIS) care permit manipularea datelor spațiale; în

acest domeniu au fost dezvoltate numeroase strategii de interogare sau structuri de date utile pentru memorarea/regăsirea eficientă a datelor.

Aplicații în domeniile chimiei și medicinei care permit manipularea și analizarea complexă de reprezentări la nivel fizic, biologic sau chimic ale organismelor vii, moleculelor organice etc.

În cele din urmă domeniul important al aplicațiilor multimedia cu ar fi sistemele pentru înregistrarea/redarea în timp real a informațiilor conținând imagini, sunet, video, unele dintre acestea cu mare impact comercial. De exemplu aplicațiile "video on demand" (selectarea unui film dintr-o bază de date de filme stocate numeric), video-conferință, prezentări multimedia pe CD-ROM, sau aplicații specifice cum ar fi sisteme de arhivare de imagini ce conțin fete sau amprente.

În sistemele de gestiune a bazelor de date folosite în aplicații ca: sisteme de informație geografică, sisteme de proiectare asistată de calculator, arhivarea sau manipularea imaginilor, sunt utilizate alături de date alfanumerice (text), date de natură grafică constând în imagini fixe (sintetizate sau digitizate), date video codificate numeric (de obicei secvențe de cadre comprimate sau nu, însoțite de informație audio sincronizată cu cea video), date de natură geometrică (puncte, linii, dreptunghiuri, poligoane simple, suprafețe, volume) sau temporală referitoare la pozițiile succesive ocupate în

spatiu (la momente diferite de timp) de obiectele componente ale bazei de date.

Caracteristicile fundamentale ale unui SGBD (caracteristici care uneori nu sunt implementate în întregime în aplicațiile din categoriile mai sus menționate) inclusiv a celor utilizate în prelucrări grafice sunt:

- *persistenta* (capacitatea entităților de a conserva informațiile caracteristice în urma diferitelor apeluri ale sistemului de programe). Din acest punct de vedere datele pot fi tranzitorii (sunt valide numai pe perioada executiei unui program/tranzactii) sau persistente (sunt stocate în fișiere pe suporturi de memorie externa și supraviețuiesc operațiilor de actualizare)
- suport pentru *existenta tranzacțiilor* (secvențe de instrucțiuni care se execută în totalitate sau deloc i.e. în mod atomic)
- *controlul concurenței prelucrărilor* (pentru a garanta consistența bazei de date în cazul executiei concurențe a mai multor tranzacții SGBD-ul impune o “ordine secvențială” de execuție i.e. chiar dacă două tranzacții se execută simultan, efectul lor este același cu cel obținut dacă tranzacțiile s-ar fi executat una după alta)
- existenta unor *sisteme de interogare* care să permită specificarea și selectarea de entități/obiecte prin intermediul unor construcții de limbaj de nivel înalt
- păstrarea *integrității* bazei de date (în urma unei tranzacții, baza de date trece dintr-o stare consistentă în alta). Consistența bazei de date este testată prin verificarea validității unor predicate (restricții de integritate) referitoare la starea curentă a bazei de date.

În plus față de caracteristicile menționate mai sus, sistemele de gestiune a bazelor de date utilizate în prelucrări grafice ar putea furniza următoarele facilități:

- suport pentru *tipuri de date spațiale* (care să permită descrierea de relații spațiale între porțiuni ale obiectelor grafice) și *limbaje de interogare spațială*.
- *interogare interactivă* care să permită rafinarea on-line a cererilor. De obicei cererile utilizator sunt exprimate prin intermediul unor interfețe grafice care oferă posibilitatea de a selecta diverse obiecte grafice și a construi cererea în mod interactiv. Mecanismele de tip QBE (Query By Example) sunt folosite adeseori.
- *extragere automată de caracteristici* ale obiectelor grafice și indexarea obiectelor în raport cu aceste caracteristici.
- folosirea *indecsilor multidimensionali și spațiali* spre deosebire de S.G.B.D.-urile clasice în care structurile de indexare se bazează pe o singură cheie
- *gruparea* obiectelor complexe. Aplicațiile care realizează gestiunea unor obiecte de date complexe (documente hipermedia, obiecte C.A.D./C.A.M.) necesită implementarea eficientă a relațiilor de incluziune sau referire reciprocă între aceste obiecte. Sunt folosite tehnici complexe de grupare a obiectelor în scopul minimizării timpului de acces (I/E) la un obiect sau a timpului necesar reconstruirii unui obiect pornind de la subobiectele care îl compun.
- metode de *memorare eficientă a masivelor binare* (BLOB – Binary Large Objects). Astfel de S.G.B.D.-uri trebuie să permită crearea și stocarea unor câmpuri binare lungi, să folosească indecsi pozitionali care să permită accesul rapid la un șir de biți începând de la o poziție dată.
- tehnici avansate de *optimizare a executiei cererilor*

2. Modele de date

Un *model de date* reprezintă o abstracție matematică a datelor și a operațiilor care pot fi efectuate asupra lor. Diferite categorii de aplicații pot lucra cu o mare varietate de formate de date; un model de date ar putea încerca să dezvolte o reprezentare unificată a acestor formate sau ar putea permite coexistența mai multor reprezentări ale datelor. Unele categorii de modele de date (interne) sunt destinate implementării de sistem iar altele (externe) sunt folosite de utilizatorii sistemelor de programe.

Modelele de date specifică modul de definire și organizare a multimilor de obiecte de date. Ele pot descrie nu numai structura fizică și implementarea obiectelor de date ci și structura logică a datelor (proprietățile logice ale valorilor de date primitive precum și modul prin care obiectele complexe sunt compuse din obiecte de date mai simple) sau comportarea datelor în procesul de calcul (structura internă a obiectelor de date este invizibilă utilizatorului fiind specificată numai comportarea funcțiilor care acționează asupra obiectelor de date).

La nivel *structural* un model de date se referă la:

- *tipurile de date primitive* care definesc o mulțime de valori primitive, operații de bază, o relație de ordine între aceste valori și o topologie peste mulțimea de valori.
- *modul de agregare* a valorilor primitive în obiecte de date compuse cum ar fi tuple de valori sau relații functionale între variabile
- *metadatele* descriu relația între date și elementele pe care acestea le reprezintă. În cazul datelor ce reprezintă mărimi fizice, metadatele ar putea desemna natura mărimii, unitățile de măsură, modul de realizare a esanționării specifice procesului de

măsurare, estimarea erorii de măsurare, valori singulare, etc.

Primele standarde în domeniul bazelor de date au fost elaborate de comitetul de limbaje CODASYL care a propus o extensie a limbajului COBOL adecvată folosirii în sistemele de baze de date. Grupul DBTG (Data Base Task Group) a elaborat în 1969 un limbaj de descriere și unul de manipulare pentru baze de date, punând astfel bazele *modelului în rețea*. Acest model pune la dispoziția utilizatorului date organizate ca tipuri înregistrare și dependente multivaloare (relații “one-many”) între aceste tipuri. Modelul în rețea permite unui tip înregistrare să fie implicat în mai multe relații.

Modelul de date ierarhic este asemănător celui în rețea dar permite unui tip înregistrare să fie implicat într-o singură relație (ca fiu al unui alt tip).

Ambele modele mai sus menționate oferă unui utilizator posibilitatea navigării printre membrii unei relații, folosind construcții de forma: `get_first()`, `get_next()`. Relațiile proprietar-membru (în cazul modelului în rețea) sau părinte-fiu (în cazul modelului ierarhic) sunt stocate implicit în înregistrările bazei de date ceea ce implică faptul că aceste modele nu asigură independența față de modul de implementare fizică a datelor. Această abordare limitează extensibilitatea, ușurința în întreținere, gradul de re folosire a componentelor și portabilitatea aplicațiilor dezvoltate folosind aceste modele.

Modelul relational introdus de T. Codd (1970) este bazat pe teoria matematică a algebrilor relationale și a calculului cu predicate de ordinul 1. Algebrele relationale cuprind operații caracteristice multimilor (intersecție, reuniune, diferență, produs cartezian) și operații relationale (selectie, proiectie, jonctiune); tipurile de

date fundamentale sunt înregistrarea (tuplul) și tabloul (relația). Acest model de date are asociat și un limbaj de interogare relational. Un prototip de astfel de limbaj este SQL (Structured Query Language) care este un limbaj declarativ spre deosebire de limbajele bazate pe navigare specifice modelelor de date prezentate anterior. S.G.B.D.-urile fundamentale care sunt bazate pe modele de tip relational sunt System/R (IBM) și INGRES (University of California at Berkeley). Alte companii producătoare de astfel de sisteme sunt Informix Corporation, Oracle Corporation, Microsoft, Borland International, Gupta Corporation etc.

O altă categorie de modele de date (postrelationale) sunt *modelele semantice* care (similar modelelor orientate spre obiecte) încearcă să modeleze cât mai apropiat de realitatea înconjurătoare. Din această categorie face parte *modelul entitate-legătură* al lui Chen (1976). În termenii acestui model o entitate este un obiect (lucru) care există și poate fi deosebit de alte obiecte (lucruri) similare. În termenii construcțiilor standard din bazele de date o entitate poate corespunde unei înregistrări iar atributele sale corespund câmpurilor înregistrării.

Modelele de date *semantice* sunt utilizate în primul rând ca mijloace de proiectare a bazelor de date. Modelarea entitate-legătură este utilizată și în analiza și proiectarea sistemelor orientate spre obiecte. De cele mai multe ori structura unei baze de date este proiectată utilizând modelul entitate-legătură și apoi este pusă în corespondență cu o schemă relatională utilizând limbajul de definire a datelor (DDL) al unui S.G.B.D. relational.

Modelele de date *functionale* încearcă includerea într-un model semantic a unor posibilități de a manipula datele utilizând "relații functionale". Astfel în modelul DAPLEX atributele sunt tratate ca funcții iar valorile atributelor sunt determinate

prin aplicarea funcțiilor corespunzătoare, unei anumite entități (sau mai multor astfel de entități). Unele sisteme de gestiune a bazelor de date orientate spre obiecte (SIM al firmei UNISYS) au la bază modele de date semantice și functionale.

Modelul de date *relational extins* menține baza teoretică solidă a modelului relational dar îi adaugă mai multă flexibilitate (semantică) permițând diferite extinderi cum ar fi includerea de tipuri de date abstracte ca și câmpuri ale unei relații, posibilitatea de a crea operații și proceduri noi cu ajutorul interfeței de programare (DPL) a bazei de date sau (în cazul modelului *obiect-relational*) posibilitatea de a include obiecte structurate, cu identitate, în câmpurile unei relații. Astfel de sisteme sunt POSTGRES (Stonebraker, Rowe, Hiohama, 1980), Intelligent SQL (Khoshafian, 1991), UniSQL și Starburst.

Comitetul de standardizare ANSI pentru limbaje folosite în cadrul sistemelor de gestiune a bazelor de date a propus în standardul SQL3 extensii cum ar fi încorporarea în limbaj a unor tipuri de date abstracte și a conceptului de mostenire. Astfel de limbaje suportă și tipuri de date adecvate aplicațiilor de prelucrare grafică a informației cum ar fi tipul BLOB ce desemnează un masiv de biți care poate conține imagini, date video sau date audio.

În modelul relational realitatea este modelată printr-o colecție de tabele, fiecare dintre ele conține un set de linii (tuple) iar câmpurile fiecărei linii (atributele tuplurilor) pot conține numai instanțe ale unor tipuri de bază, atomice (integer, float, string). Această reprezentare este cunoscută sub numele de "prima formă normală". Modelele relationale imbricate (Schek, Scholl, 1986) permit utilizatorului să descrie un atribut a cărui valoare să fie o relație (multime de tuple) și deci să reprezinte/manipuleze direct mulțimi întregi de tuple. Se creează astfel spații de obiecte, având structură arborescentă,

situate la un nivel superior față de colecțiile de tipuri atomice de bază.

Modelele bazate pe obiecte complexe permit nu numai imbricarea relațiilor ci și a unor obiecte arbitrare. Obiectele complexe pot fi construite prin utilizarea a doi constructori “sets” și “tuples”. Aceștia permit crearea de spații de obiecte de forma unui arbore; există însă modele care permit ca același obiect să fie un subiect al mai multor “obiecte părinte”, astfel de spații de obiecte sunt structurate sub forma unui graf. Pentru a putea suporta această structură a spațiului obiectelor, modelele orientate obiect trebuie să permită conceptul de identitate a unui obiect (un obiect poate fi referit printr-un număr unic, generat intern, independent de valoarea cheii sale primare – care poate să existe sau nu).

În funcție de nivelul la care se face abstractizarea, modelele de date pot fi grupate în trei categorii: fizic, de implementare și conceptual.

Modelele la *nivel fizic* nu au în general suficientă flexibilitate pentru a putea cuprinde toate caracteristicile comune și operațiile aferente datelor din categorii cât mai largi de aplicații grafice. Astfel de modele pot fi utile în standardizarea transferului de date între diferite aplicații.

Categoriile cele mai des folosite de modele de date (la nivel de implementare) sunt modelele: relational, ierarhic, în rețea și orientat spre obiecte. Descrierea unei baze de date la nivel de implementare alcătuiește așa numita schemă de implementare a bazei de date. La nivelul implementării unei aplicații, modelele de date cele mai adecvate prelucrărilor grafice sunt *modelul orientat spre obiecte* și *modelul relational extins*. Astfel de modele pot fi dezvoltate pentru a corespunde unor categorii relativ specializate de aplicații (de exemplu aplicațiile din domeniul vizualizării științifice).

La nivel conceptual au fost propuse câteva categorii de modele de date independente de implementare, un exemplu ar fi modelul bazat pe teoria laticelor iar un altul este modelul “fasciculelor de fibre” folosit în aplicațiile de vizualizare științifică a câmpurilor vectoriale.

În aplicațiile de natură grafică un rol esențial îl au și metadatele care descriu informația sintactică și semantică despre date, structura, atributele și istoricul manipulării acestora. Metadatele structurale descriu tipul și structura datelor (schema bazei de date) pentru a putea fi folosite de mecanismele de acces la date. O altă categorie de metadate se referă la relațiile dintre date și entitățile pe care acestea le reprezintă (de exemplu numele câmpurilor de date și domeniile de variație ale valorilor de date). Metadatele de documentație pot reprezenta comentarii referitoare la date sau proceduri pentru descrierea unor prelucrări ce se vor executa la momente ulterioare de timp.

3. Modele de date în baze de date spațiale

Pentru a proiecta o bază de date spațială trebuie avut în vedere modul în care sunt integrate reprezentările datelor spațiale respectiv nespațiale. Din acest punct de vedere pot exista sisteme dedicate, duale, stratificate, integrate.

Sistemele dedicate corespund unor aplicații strict specializate, ele nu prezintă facilități complete caracteristice bazelor de date. Astfel de sisteme nu sunt extensibile și nu pot realiza acțiuni care nu au fost gândite anterior de către proiectanții lor. Un exemplu tipic sunt sistemele de proiectare asistată de calculator în domeniul electronicii (RACAL-REDAC, Cadence, ELECTRO-CAD). Astfel de sisteme arhivează date geometrice necesare documentației de proiectare și execuție a

plăcilor cu circuite imprimare, permit amplasarea automată/manuală a componentelor și rutarea traseelor de interconexiune cu respectarea unor reguli de proiectare.

Arhitecturile de sistem duale disting datele spațiale de cele nespațiale prin utilizarea unor modele de date distincte. O astfel de arhitectură implică existența unor module diferite de gestiune a memoriei. Comunicatia între cele două subsisteme (spațial/nespațial) se face cu ajutorul unor identificatori comuni. Aplicațiile realizate conform acestei arhitecturi ar putea avea dezavantaje legate de sincronizarea accesului la date sau de păstrarea integrității informației geometrice. Astfel, acțiunile efectuate asupra porțiunii spațiale a datelor ar putea să nu fie reflectate în porțiunea nespațială. Un exemplu de S.G.B.D. spațial cu arhitectură duală este sistemul de informație geografică ARC/INFO.

Existența a două module distincte de gestiune a memoriei ar putea fi evitată prin memorarea datelor spațiale conform unui model de date relational. În acest caz datele spațiale trebuie transformate (de exemplu reprezentate prin puncte caracteristice) sau descompuse în părți constitutive (de exemplu interiorul unei regiuni este descompus în mai multe blocuri sau în pixeli). Această abordare stă la baza *arhitecturii stratificate*. Ierarhia de organizare a straturilor ar putea fi: nivelul de bază al S.G.B.D.-ului relational, nivelul spațial și nivelul aplicației propriu-zise (care conține rutine specifice aplicației). Exemple de baze de date realizate conform acestei arhitecturi sunt SIRO-DBMS și GEO-VIEW.

Arhitecturile integrate permit utilizatorului să extindă o bază de date cu propriile sale tipuri abstracte de date adecvate aplicațiilor spațiale. Pentru realizarea implementării unei astfel de arhitecturi cele mai adecvate ar fi modelul de date

relational extins sau cel orientat spre obiecte. Aceste sisteme permit adăugarea unor construcții ca: tipuri de date abstracte, obiecte complexe, câmpuri procedurale sau atribute multivaloare. Fiecare din tipurile de date (spațiale sau nu) trebuie reprezentate printr-o structură de date corespunzătoare care să îndeplinească cel mai bine cerințele problemei. Exemple de sisteme de gestiune a bazelor de date ce respectă o astfel de arhitectură sunt: Gral, POSTGRES, Probe, GEO++.

O problemă importantă în sistemele bazate pe o arhitectură integrată este modul de a realiza legăturile între descrierea datelor spațiale asociate unui obiect complex și restul informației nespațiale a obiectului. O opțiune este cea realizată în sistemul GEOQL unde fiecare relație are asociat un singur atribut spațial. Sistemul acordă o importanță mai mare componentei relationale decât celei spațiale; astfel și atributele spațiale ale relațiilor sunt implementate folosind structuri de date avansate (indecsi ierarhici multidimensionali) pentru a efectua operații spațiale asupra a două câmpuri (din două relații distincte) cu informație spațială va trebui construită o nouă relație (intermediară) ceea ce va micșora eficiența execuției cererilor spațiale.

În alte sisteme datele spațiale și cele nespațiale se referă reciproc prin legături bidirectionale. Fiecărui atribut spațial din schema bazei de date îi este asociată o structură de date spațială utilizată pentru a stoca toate instanțele aceluia atribut spațial, aferente unei mulțimi de obiecte. Această structură de date constituie un index pentru obiectele spațiale având un anumit tip (punct, linie, regiune) precum și un context asupra căruia se pot executa diferite operații spațiale (transformări geometrice, intersecții de poligoane, calcule de distanță etc.)

Ideea de a introduce noi tipuri de date abstracte ca domenii asociate unor atribute

într-un S.G.B.D. relational și cea de a permite utilizatorului să-și definească propriile sale structuri de indexare se datorează lui Stonebraker. Conform acestei abordări folosite în multe S.G.B.D.-uri extensibile structurile de date spațiale sunt considerate structuri de indexare a informației.

Construirea unei baze de date spațiale ar putea avea la bază și un model orientat spre obiecte. Astfel se asigură un grad înalt de structurare a informației, prin introducerea conceptelor de clasă și mostenire. Independența reprezentării fizice și a celei logice a datelor și manipularea comodă a lor sunt facilitate prin încapsularea datelor și supraîncărcare. Tipurile de date spațiale pot fi implementate prin clase iar conceptul de mostenire este utilizat pentru definirea de subclase. Un exemplu de astfel de sistem a fost descris de Bancilhon (S.G.B.D.-ul orientat spre obiecte O2). În acest sistem datele spațiale sunt modelate pe două nivele: geometric și “de mapare”. O bază de date spațială este alcătuită dintr-o multime de mapări iar o mapare este o relație care are cel puțin un atribut spațial. Nivelul geometric corespunde atributelor spațiale care sunt reprezentate prin tipuri de date abstracte geometrice cum ar fi tipul punct, linie sau regiune.

Bibliografie:

[Banc91] F. Bancilhon, L. Delobel, P. Kanellakis, Building an Object-Oriented Database System: The Story of O₂, Morgan Kaufman, San Mateo California, 1991.

[Catt94] R. G. Cattell, Object Data Management, Addison Wesley Publishing Company, 1994.

[Egen94] M. J. Egenhofer, Spatial SQL: A Query and Presentation Language, IEEE Transactions on Knowledge and Data Engineering, february 1994.

[Gallo94] J. Gallop, Underlying Data Models and Structures for Visualization; apărut în: Scientific Visualization, Advances and Challenges, Academic Press, 1994.

[Gupta91] A. Gupta, T. Weymouh, R. Jain, Semantic Queries with Pictures: The VIMSYS Model, Proceedings of the 17-th International Conference on Very Large Databases, 1991.

[Mai91] R. Maiocchi, B. Pernici, Temporal Data Management Systems: A Comparative View, IEEE Transactions on Knowledge and Data Engineering, december 1991.

[Oren88] J. A. Orenstein, F. A. Manola, PROBE Spatial Data Modelling and Query Processing in an Image Database Application, IEEE Transactions on Software Engineering, may 1988.

[Samet94] H. Samet, Spatial Data Models and Query Processing, ACM Press, 1994.