

Sisteme de interogare folosite în bazele de date pentru prelucrări grafice

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

Lucrarea prezinta câteva categorii de cereri specifice bazelor de date pentru prelucrări grafice. Sunt date exemple din limbajele de definire a datelor sau limbajele de specificare a cererilor asociate unor sisteme de gestiune a bazelor de date utilizate cu precadere în domeniul aplicatiilor grafice.

Cuvinte cheie: bază de date, baza de date spatia, sistem de interogare, model de date.

1. Categoriile de cereri specifice bazelor de date pentru prelucrări grafice

Sistemele de interogare ale bazelor de date folosite în prelucrări grafice sunt caracterizate prin următoarele elemente: specificarea cererilor poate conține predicate de cautare vagi pentru identificarea unor entități cu caracteristici similare altora; există cereri ce se referă la atributele tipurilor grafice (culoare, textură), cereri spațiale, cereri vizuale bazate pe exemple sau liste de selecție interactivă (pick list), cereri în care se asociază ponderi (priorități) predicatelor componente. În prelucrarea cererilor, mai ales în cazul celor dependente de context sau al celor de similitudine, faza de optimizare a execuției este deosebit de complexă. Rezultatele unei cereri (dacă sunt mai multe) sunt ordonate în funcție de ponderile asociate termenilor cererii sau valorilor atributelor obiectelor din baza de date. Spre deosebire de sistemele de baze de date conventionale în care după obținerea rezultatelor unei cereri procesul de interogare ia practic sfârșit, în unele sisteme avansate de regasire a informației apare conceptul de "feedback de relevanță" (pentru utilizator), reacția de acceptare sau nu a rezultatelor cererii de către utilizator este folosită pentru a recompune cererea și a o supune din nou spre rezolvare sistemului de interogare.

Cererile în sistemele de baze de date spațiale au în general nivele de abstractizare diferite. La nivelele superioare, interogările cele mai frecvente se referă la detectarea unor sabloane în datele de analizat sau la anticiparea caracteristicilor datelor situate într-o anumită regiune sau la un anumit moment de timp. La nivele

inferioare de abstractizare cererile specifice ar putea realiza: conversii între diferite formate de date, schimbarea caracteristicilor de vizualizare, îmbunătățirea unor parametri ai imaginilor.

Cererile situate la un nivel de abstractizare intermediar pot fi rezolvate cel mai bine folosind sisteme de gestiune a bazelor de date. Aceste cereri consideră datele ca fiind organizate ierarhic (în straturi). Fiecare strat este partitionat în zone (colecții de locații cu valori comune ale atributelor). De exemplu în cazul sistemelor de proiectare asistată de calculator a plăcilor cu circuite imprimate, un strat ar corespunde traseelor iar altul pașilor sau gaurilor de trecere iar într-un sistem de informație geografică straturile ar putea descrie modul de utilizare al parcelelor de teren, tipul de sol, rețeaua de drumuri etc.

În astfel de sisteme în care datele sunt organizate pe nivele, cererile pot fi clasificate ca: *cereri locale* (acestea se referă la caracteristici aflate la aceleași locații pe niveluri diferite) *cereri zonale* (se referă la grupuri de locații situate pe același nivel care au aceeași valoare de atribut) *cereri de focalizare* (se referă la vecinătățile unei locații situate pe un anumit nivel, vecinătatea poate fi delimitată restricții referitoare la distanță, direcție sau timp). Această ultimă categorie de cereri este echivalentă cu *cererile de domeniu* (range queries) din sistemele de baze de date conventionale (la aceste cereri domeniul în care trebuie să se afle punctul de date este un hiperdreptunghi).

O categorie importantă de cereri specifice bazelor de date spațiale sunt cele de *suprapunere poligonală*; în această situație sunt suprapuse două straturi caracterizate prin atri-

bute spatiale diferite si, în fiecare locatie spatia, este aplicata o functie asupra valorilor atributelor corespunzatoare. Daca se face o analogie între "straturile cu informatii" dintr-o baza de date spatia si relatiile unei baze de date obisnuite, rezultatul unei cereri de tip suprapunere de poligoane este analog cu cel obtinut prin efectuarea produsului cartezian al multimilor având ca elemente regiunile celor doua straturi cu informatii.

Cererile zonale pot fi folosite pentru a implementa cazuri particulare de cereri de suprapunere poligonala. De obicei o cerere zonala lucreaza cu trei straturi de informatie spatia, primele doua constituie operanzii cererii iar al treilea va contine rezultatul. Informatia din primul strat (operand) actioneaza ca o masca ce va determina partitionarea celui de-al doilea strat în zone. Stratul al treilea va contine rezultatul aplicarii cererii zonale tuturor locatiilor din al doilea strat care corespund fiecarei zone de interes. De exemplu, în cazul unei baze de date spatiale ce retine date referitoare la culturile agricole dintr-o anumita zona geografica, o astfel de cerere ar putea fi formulata astfel: "Sa se determine cantitatea de apa pluviala medie anuala în zonele unde creste grâu Triticum Durum". Ea ar corespunde operatiei de intersectie între stratul de informatie ce se refera la cantitatea de apa/m²/an si stratul de informatie referitoare la zonele în care creste grâu Triticum Durum. În final se va efectua un calcul de mediere de forma $\sum_i f_i A_i / \sum_i A_i$, unde sumarea se face peste toate zonele unde creste grâu T.D. si care au o singura valoare a indicatorului "precipitatie medie anuala" egala cu f_i ; A_i reprezinta aria unei astfel de zone.

2. Limbaje de interogare specifice bazelor de date pentru prelucrari grafice

În proiectarea si dezvoltarea unui limbaj de interogare al bazelor de date spatiale au fost utilizate abordari bazate pe extinderea unor limbaje ca SQL (în care structura fundamentala de interogare este organizata sub forma unui sablon SELECT ... FROM ... WHERE, clauze care corespund respectiv operatiilor de proiectie, produs cartezian si selectie, caracteristice algebrilor relationale) QUEL

(limbaj bazat pe calculul relational cu tuple în care clauzele folosite sunt RANGE, FROM, WHERE) sau QBE (limbaj de interogare bazat pe exemple).

În limbajele de interogare specifice bazelor de date spatiale, conceptele si reprezentarile sub forma tabulara a datelor, caracteristice bazelor de date relationale nu sunt preponderente, un rol mai important îl joaca (datorita naturii geometrice a unei parti a datelor) dezvoltarea unor metode simbolice de reprezentare (icone) sau a altor metode de interactiune. De aceea, în contextul bazelor de date spatiale, limbajul de interogare trebuie sa permita nu numai regasirea unor informatii ci si descrierea unui mod adecvat de prezentare a datelor spatiale.

Un sistem de interogare pentru o baza de date spatia trebuie ca, în afara caracteristicilor specifice limbajelor clasice de interogare, sa satisfaca urmatoarele conditii:

- posibilitatea de a defini un *tip abstract de date spatia* pentru ca utilizatorii sa poata referi datele spatiale la un nivel independent de codificarea interna a acestora
- rezultatul unei cereri trebuie sa poata fi afisat în forma grafica (folosind attribute de culoare, intensitate, simboluri, sabloane de hasura) care este adecvata pentru analiza datelor spatiale si vizualizarea caracteristicilor obiectelor spatiale ale bazei de date care sunt reprezentate prin attribute nespatale;
- prezentarea grafica a informatiei trebuie sa permita si afisarea *contextului spatia* (i.e. o informatie a carei afisare nu a fost ceruta în mod special dar care este necesara pentru a putea interpreta corect rezultatul cererii);
- trebuie sa existe mecanisme de control al utilizatorului asupra continutului unui desen de prezentare a datelor spatiale
- corespondenta între clasele obiectelor spatiale si reprezentarile lor grafice este realizata prin intermediul unei "legende" grafice si a unor etichete asociate obiectelor spatiale
- mecanismele de vizualizare a datelor spatiale trebuie sa permita determinarea dimensiunii reale a obiectelor desenate precum si restrângerea zonei de cautare a unei informatii.

2.1. Extinderea limbajelor de interogare cu

facilitati orientate spre obiecte

Dupa modelul unor limbaje de programare consacrate C, LISP, PASCAL care au primit extensii orientate spre obiecte (C++, CLOS sau Borland PASCAL), limbajul de interogare/manipulare a datelor SQL a primit extensii orientate spre obiecte. Astfel, standardul SQL3 introduce caracteristici specifice bazelor de date avansate cum ar fi: extensii OO, extensii pentru specificarea restrictiilor de integritate, suport pentru recursivitate, puncte de salvare intermediare.

Alaturi de tipurile de date predefinite (INTEGER, FLOAT, CHARACTER, DATE, TIMESTAMP) au aparut constructii de limbaj pentru specificarea tipurilor abstracte de date definite de utilizator. De exemplu, constructia

```
CREATE TYPE imagine
WITH OID
UNDER grafica
( PRIVATE
  size INTEGER,
  resolution FLOAT,
  content PCX,
PUBLIC
  CONSTRUCTOR PROCEDURE
    imag(I imagine, cale VARCHAR);
  DESTRUCTOR PROCEDURE
    d_imag(I imagine);
  ACTOR FUNCTION
    get_size(I imagine) returns integer;
  ...
);
```

specifica numele tipului abstract de date, faptul ca fiecare obiect instantia al acestui tip va avea un identificator unic generat de sistem; obiectul poate fi referit prin acest identificator. Clauza PUBLIC este analoaga celei din C++ (atributele unei instante TAD pot fi PUBLIC, PRIVATE sau PROTECTED). Clauza UNDER specifica ierarhia de tipuri (faptul ca TAD-ul "imagine" mosteneste de la TAD-ul "grafica"). Este posibila mostenirea multipla i.e. un tip poate prelua proprietati de la mai multe alte tipuri.

Standardul defineste trei categorii de rutine care pot fi asociate unui TAD: constructorii (initializeaza instante ale tipului abstract de date), destructorii (elibereaza resurse alocate

unor instante ale TAD-ului) si actorii (care îndeplinesc orice alta operatie asupra instantelor TAD-ului). Odata definit, un tip abstract de date poate fi folosit în declaratii de tabele CREATE TABLE specifice limbajului de definire a datelor (DDL). In SQL3 baza de date contine la nivelul superior numai tabele (relatii), instantele TAD-urilor nu pot fi folosite decât pentru definirea câmpurilor unor relatii sau a argumentelor unei rutine.

Standardul ISO/IEC/WG3 defineste tipuri de date abstracte pentru a extinde sistemul SQL în domeniul bazelor de date multimedia (SQL/MM) [Khos 96]. Astfel definitiile TAD-urilor formeaza o ierarhie de clase având ca radacina tipul GEO_GeometricObject; unele dintre subclasele importante sunt GEO_PointTypes, GEO_LineTypes, GEO_AreaTypes, GEO_VolumeTypes, GEO_RasterTypes, GEO_GraphTypes sau GEO_TextTypes.

De exemplu, în SQL3, o celula 3D se defineste ca o subclasa a clasei GEO_VolumeClass:

```
CREATE TYPE GEO_Celula_3D
UNDER GEO_VolumeClass,
(forma GEO_Cell_3DDefinition, __ poate
  __ fi paralelipedica
  pos GEO_Cell_3DReference NOT NULL,
  __lista vecinilor celulei

  xDim REAL,
  yDim REAL,
  zDim REAL
)
```

O alta abordare pentru a realiza integrarea mecanismelor de baze de date si a celor de programare orientata spre obiecte este *extinderea limbajelor de programare OO cu mecanisme specifice bazelor de date*, aceste extensii pot include facilitati ca: interogarea unor baze de date, suport pentru tranzactii sau persistenta. De exemplu limbajul OPAL (sistemul GemStone) a extins Smalltalk cu clase referitoare la gestiunea bazelor de date, alte sisteme dezvoltate în aceeași maniera sunt Versant (SGBD orientat spre obiecte al firmei Versant Object Technology) si Object Store. Unii producatori (Microsoft, SUN, Objectivity, ServioLogic) au extins si limbajul C++ cu concepte caracteristice bazelor de date orientate spre obiecte. Alaturi de cunoscutele

caracteristici: tipuri de date abstracte (clase), mostenire (simpla si, în versiuni recente, multipla), polimorfism si legare dinamica (prin posibilitatea supra-încărcării numelor de functii sau operatorilor) sau mai recent suportarea de tipuri para-metrizate, au fost introduse noi facilitati ca:

- posibilitatea definirii de clase persistente prin mostenirea caracteristicii de persistenta de la o clasa radacina care are metode specifice care sa permita crearea unor noi instante si stocarea obiectelor persistente (new), distrugerea unui obiect persistent, citirea câmpurilor unui obiect din baza de date si scrierea starii curente a obiectului în baza de date, zăvărârea accesului la obiect pentru a realiza controlul explicit al concurenței de către utilizator.
- definirea de extensii (o extensie este mulțimea tuturor obiectelor instanțe ale unei clase) si specificarea explicita de către programatorul de aplicatie dacă sistemul OODB va mentine sau nu aceasta extensie. În caz afirmativ, componentele extensiei vor putea fi vizitate si manipulate folosind programe C++.
- definirea de colecții de obiecte persistente (tablouri, mulțimi, liste si dictionare de obiecte) si relatii. Instanțele acestor "obiecte colecție" vor fi valori ale unor variabile instanța dintr-o clasa persistenta. Exista mecanisme de inserare, stergere sau actualizare a unui obiect dintr-o colecție de obiecte. De asemenea, obiectele colecției au asociati "iteratori" i.e. functii pentru a parcurge elementele colecției.

Dupa mai multi ani de dezvoltare independenta de multe ori fara sa acorde atentie interactiunii si interoperabilitatii, o serie de producatori de baze de date orientate spre obiecte au elaborat standardul ODMG (Object Database Standard). Companiile participante la dezvoltarea acestui standard sunt Ontos, Versant, O2, Objectivity, Hewlett Packard, POET Software, Intellitic, DEC, Servio, Texas Instruments. Componentele fundamentale ale acestui standard sunt: modelul de date ODMG-93, limbajul de definire a obiectelor (ODL), limbajul de interogare a obiectelor (OQL), legaturi cu limbaje de programare (C++).

Modelul de date ODMG este bazat pe

obiecte si permite specificarea identificatorilor de obiect. Categoriile de obiecte sunt grupate în tipuri (obiectele unui tip au o comportare comuna si un domeniu comun de stari), comportarea obiectelor este definita prin operatii iar starea obiectelor este definita printr-o multime de proprietati (attribute ale obiectului sau relatii între un obiect si unul sau mai multe alte obiecte). Obiectelor li se pot da nume (un obiect poate avea mai multe nume dar un singur identificator).

În figura 1 este prezentata arhitectura unei baze de date realizata în conformitate cu standardul ODMG. Aplicatia rezulta prin combinarea descrierilor ODL si OML cu rutinele puse la dispozitie de SGBD, rutine care permit exploatarea bazei de date. Programul sursa este scris într-un limbaj de programare (de obicei C++) extins cu primitive OML inclusiv cele ce permit utilizarea tranzactiilor sau specificarea de cereri. Declaratiile ce alcatuiesc schema bazei de date pot fi scrise în ODL sau într-o extensie a limbajului de programare în care este realizata aplicatia.

Exista tipuri de date colectie: mulțimi, liste si tablouri. O colecție poate contine un numar arbitrar de elemente care sunt toate instanțe ale aceluiași tip. Instanțele (cu nume) acestor tipuri pot fi utilizate pentru a grupa obiecte. Tipurile obiect sunt legate într-un graf subtip/supertip. Pot fi declarate extensii care sa contina toate instanțele unui tip de date. În anumite cazuri, instanțele individuale ale unui tip de date pot fi identificate în mod unic prin chei formate din una sau mai multe proprietati ale obiectului. Colecțiile si structurile pot fi compuse în mod arbitrar. Modelul ODMG suporta mulțimi de structuri, structuri de mulțimi, tablouri de structuri etc.

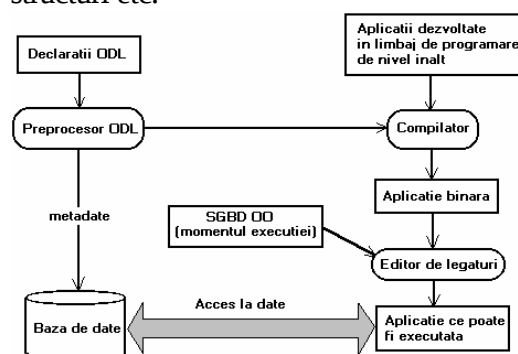


Fig. 1. Arhitectura unei Baze de Date ODMG
Limbajul de definire a obiectelor (ODL) cuprinde construcțiile sintactice necesare specificării caracteristicilor menționate anterior ale modelului de date. Acest limbaj permite ca o aplicație să fie portată într-un nou limbaj de programare fără să se rescrie schema de date. El este destinat definirii unor tipuri de obiect care pot fi implementate în diferite limbaje de programare. Sintaxa ODL se bazează pe limbajul de definiție de interfață (IDL) elaborat de grupul CORBA. O schema specificată în ODL poate fi implementată în orice SGBD orientat spre obiecte care suportă standardul ODMG. ODL-ul poate fi folosit în descrierea datelor și operațiilor unei aplicații, independent de limbajul de programare.

3. Exemple de sisteme de interogare în bazele de date pentru prelucrări grafice

3.1. Limbajul Spatial SQL

Limbajele din această categorie se bazează pe SQL, ele extind gramatica SQL cu o mulțime de operatori spațiali și relaționali care se referă la date spațiale. Unele limbaje cum ar fi PSQL (dezvoltat de Roussopoulos) nu permit adăugarea de noi operatori la momentul formulării cererilor.

O altă abordare permite adăugarea unor funcții (operatori) definite de utilizatori care sunt făcute cunoscute sistemului de interogare la momentul execuției cererilor. Astfel de funcții pot avea valori scalare (de exemplu arie sau perimetrul unei regiuni) sau pot fi dependente multivaloare (care au ca rezultat relații). Unii autori numesc aceste din urmă dependente, funcții tabelare. Spre exemplu în limbajul de interogare *Spatial SQL*, Egenhofer propune ca datele spațiale să fie abstractizate prin extinderea domeniului calculului relațional cu alte patru domenii spațiale numite *spatial0*, ..., *spatial3* în funcție de dimensiunea obiectelor spațiale reprezentate. Un atribut al domeniului spațial se numește atribut spațial iar o relație ce se referă la cel puțin un atribut spațial se numește relație spațială. Astfel de relații au proprietăți diferite față de cele ale relațiilor obișnuite (cu câmpuri de tip numeric sau sir de caractere) ele se referă la concepte topologice sau metrice și sunt

folosite similar cu relațiile convenționale.

Operatorii spațiali pot fi *unari*, de exemplu: dimensiune (determină dimensiunea topologică a unui obiect spațial), frontieră (determină "fetele" frontierei unui obiect *n*-dimensional), interior (determină părțile unui obiect care nu sunt situate pe frontieră sa). Tot din categoria operatorilor spațiali unari fac parte operatorii: lungime, arie și volum care exprimă proprietățile aritmetice ale unor obiecte spațiale, precum și operatori ce combină aspectele aritmetice cu cele topologice: perimetru, muchii frontieră, înfasurătoare convexă, coordonate extreme.

Operatorii spațiali binari calculează o valoare pornind de la două tuple ale unor relații spațiale, de exemplu operatorii:

distanță: atribut_spațial x atribut_spațial →

real

direcție: atribut_spațial x atribut_spațial →

0 .. 359° 59' 59"

Operatorii relaționali spațiali binari descriu relații între două atribute spațiale și produc rezultate booleene; de exemplu operatorul "conține" (o linie poate conține sau nu un punct). Relațiile binare topologice se bazează pe determinarea intersecțiilor între frontierele sau interioarele a două regiuni. Operatori de acest tip se pot aplica oricărui două obiecte spațiale de dimensiuni arbitrare. Ca exemple de astfel de operatori se pot menționa: *e_disjunct*, *intersectează*, *se_suprapune*, *conține*, *acopera*, *e_egal*. Se pot specifica și relații de ordine spațială de tipul: stânga/ dreapta, nord/sud sau sus/jos.

O altă caracteristică a bazelor de date relaționale care trebuie adaptată, extinsă cu noi facilități în cazul bazelor de date spațiale este limbajul de definire al datelor (DDL). Majoritatea sistemelor de baze de date spațiale bazate pe SQL permit specificarea de atribute spațiale în clauza CREATE TABLE. Astfel Egenhofer folosește numele de atribut *geometry* având tipul *spatial_n* ($n \in \{0,1,2,3\}$) și consideră ca o relație spațială poate avea în general un singur atribut spațial care să definească geometria unui obiect. Există posibilitatea ca un obiect să aibă mai multe geometrii ceea ce ar putea fi util în cazul prezentării la diferite nivele de detaliu a

respectivului obiect.

Samet propune ca tipuri de date disponibile pentru atributele spatiale: LINE_SEGMENT, REGION, POINT, POLYGON, BOX. Atributele spatiale si cele nespatale sunt plasate la acelasi nivel conceptual, valoarea unui atribut spatial fiind comuna tuturor atributelor nespatale ale unui tuplu al unei relatii.

Operatiile standard de manipulare a datelor: proiectia, selectia, jonctiunea caracteristice domeniului nespacial sunt adaptate si pentru domeniul spatial. Samet caracterizeaza un predicat (specificat în clauza WHERE) ca fiind spatial daca se refera la cel puțin un atribut spatial sau foloseste un operator spati-al; în caz contrar predicatul este relational.

Dintre operatiile de interogare specifice bazelor de date spatiale se remarca selectia grafica interactiva, specificata printr-o cerere de forma: SELECT nume FROM oras

WHERE geometry=PICK;

Semantica cererii de selectie grafica interactiva depinde de dimensiunea topologica a obiectului indicat. In procesul de selectie pot apare ambiguitati daca cel puțin doua obiecte din clasa care trebuie identificata sunt situate la aceeasi distanta de pozitia cursorului selector, în astfel de cazuri utilizatorului trebuie sa i se ofere mai multe variante de alegere pentru a identifica obiectul dorit.

3.2 Limbajul de cereri si modelul de date în sistemul VIMSYS

Modelul de date utilizat în sistemul VIMSYS [Gupta 91] (Visual Information Management System) (model stratificat) este orientat spre obiecte; el grupeaza entitatile de informatie în 4 plane diferite, acestea sunt denumite DO (Domain Objects and Relations) DE (Domain Events and Relations) IO (Image Objects and Relations) IR (Image Representations and Relations). Toate obiectele au asociate atribute si metode, atributele sunt legate într-o ierarhie. Relatiile pot fi spatiale, structurale sau semantice.

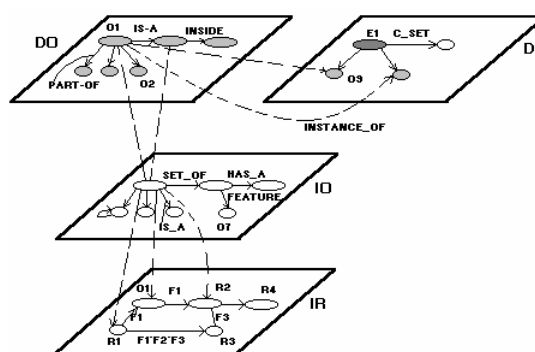


Fig. 2. Modelul de date VIMSYS

Datele constau în imagini, fiecare având asociate un antet precum si o înregistrare cu câmpuri alfanumerice. Elementele constitutive ale modelului de date sunt: reprezentarile (i.e. tipurile de date pe baza carora sunt construite obiectele) char, int, float, bool, string; constructorii set_of, tuple_of, vector_of, matrix_of, graph_of, sequence_of. O reprezentare de imagine este un tip abstract de date; pentru fiecare din obiectele imagine din planul IO, pot exista mai multe reprezentari asociate aceluiași obiect. In cursul procesului de interogare utilizatorul foloseste o interfata grafica si un limbaj natural cu restrictii pentru a specifica semantica unor caracteristici ale sistemului.

Sistemul dezvoltă o metodologie de organizare si indexare a imaginilor astfel încât sa fie posibil accesul complet sau partial la matricea unei imagini. Caracteristicile imaginilor sunt ordonate într-o ierarhie ceea ce permite construirea unor caracteristici mai complexe pornind de la unele mai simple. Modelul de date trebuie sa permita utilizarea datelor spatiale si a fisierelor.

In planul IO exista trei clase principale de obiecte: imagini, caracteristici de imagini, organizarea caracteristicilor; aceste obiecte sunt legate prin relatiile set_of, is_a si feature_of. Localizarea regiunilor în imagini este realizata prin indexare folosind arborii R^+ . Dintre caracteristicile pe care le poate avea o imagine sunt implementate textura (se refera la densitatea si orientarea muchiilor), culoarea, intensitatea si caracteristici geometrice (puncte, linii).

Planul DO contine o descriere la nivel semantic bazata pe informatii continute în planele IO si IR. Obiectele din planul DO sunt conectate printr-un graf (Object Relation Graph). Un

obiect DO poate fi o subclasa a unui obiect (sau mai multor obiecte) din planul IO. De exemplu, daca obiectele O_1 , O_2 situate în planul DO sunt subclase (sau obiecte proto-tip) ale obiectelor ω_1 , ω_2 situate în planul IO, perechea (O_1 , O_2) va reflecta toate relatiile între ω_1 si ω_2 . Un arc (sursa, destinatie) între doua obiecte situate în planul DO semnifica faptul ca obiectul sursa are printre metodele sale o functie care poate genera obiectul destinatie. Pentru aceasta este suficient ca obiectul destinatie sa trimita un "mesaj cerere" cu parametri corespunzatori.

Planul DE este inclus în modelul de date pentru a permite specificarea unor evenimente ce pot apare în timpul derularii unor secvente de imagini. Un eveniment (în acest context) este un ansamblu de caracteristici colectate asupra unei secvente de imagini. Evenimentul "rotatie de imagini" va fi reprezentat printr-o multime de regiuni de imagini, un atribut (unghiul de rotatie), o multime de restrictii, si de metode care sa gaseasca regiunile care satisfac respectivele restrictii.

Sistemul de interogare permite accesul la o instanta a unei imagini prin specificarea unei cereri referitoare la orice multime de obiecte (care pot fi situate în mai multe planuri). În formularea unei cereri sunt posibile unele ambiguitati astfel: utilizatorul poate folosi numele unui obiect semantic care nu este definit în sistem (va trebui sa îl defineasca mai târziu pe baza obiectelor din sistem), sunt acceptate cereri de similitudine. Interfata utilizator este bazata pe meniuri, forme, butoane de selectie si ferestre multiple text sau imagine. O fereastră imagine permite executarea unor operatii specializate ca: selectarea unei portiuni de imagine, a unei regiuni, linie sau punct precum si a unor operatii interne ca: panoramare, scalare, modificarea unor atribute de afisare.

În sistemul VIMSYS se propune un model de interogare în care utilizatorul nu specifica explicit clauza SELECT ci incremental. Sistemul utilizeaza o interfata grafica pentru realizarea interogarii, o baza de cunostinte pentru a asista utilizatorul în procesul de detaliere a unei cereri si un dictionar de date pentru a stoca statistici referitoare la baza de date. Pentru a formula o cerere, utilizatorul specifica domeniul de interes

urmat de aspectul de interes din cadrul domeniului (un aspect este o vedere a schemei OAR - Object Attribute Relation - având un nume unic). Modulul de cunoastere primeste la intrare o combinatie (domeniu, aspect) si determina un subgraf al structurii OAR în planele DO si DE. Utilizatorul va alege obiectele (relatiile) care sunt necesare în cererea sa, modulul de cunoastere va analiza obiectele în raport cu contextul (domeniu, aspect) dorit si va afisa multimea atributelor relevante ale fiecărei activitati selectate. Utilizatorul va alege o submultime a atributelor asociind pentru fiecare din acestea un predicat.

3.3. Limbajul de interogare Gral (algebre multisort)

Una din optiunile de proiectare a unui Sistem de Gestiune a Bazelor de Date care sa poata gestiona obiecte complexe este daca structura acestor obiecte sa fie sau nu vizibila la nivelul modelului de date. Un obiect complex ar putea fi descris structural ca o colectie de tuple ale mai multor relatii sau comportamental ca un simplu atribut care ia valori într-un domeniu specific respectivului obiect. În construirea SGBD-urilor extensibile au fost urmate doua directii: definirea cu precizie a unui model de date pentru sistem, ca si a interfetei cu utilizatorul asociata (POST-GRES) sau punerea la dispozitia utilizatorului a unei colectii de unelte de programare ce ar putea fi folosite în construirea unui SGBD, în acest din urma caz utilizatorii bazei de date pot realiza direct adaptarea sau chiar implementarea unor portiuni ale sistemului.

Un sistem extensibil din a doua categorie Gral [Güt 89] permite utilizatorului sa adauge tipuri de date sau operatii noi la limbajul de interogare, sa defineasca reprezentarea unui tip sau sa adauge noi structuri de indexare care sa fie adecvate noilor tipuri de date sau reprezentari de relatii. Este posibila si adaugarea unor operatii specifice noilor tipuri de date sau specificarea unor reguli pentru a realiza optimizarea implementarii operatiilor de interogare adaugate de utilizator.

Limbajul de interogare al sistemului Gral face parte din clasa limbajelor algebrice. Flexibi-

litatea deosebita a limbajului de cereri se datoreaza utilizarii *algebrelor multisort* care permit atât specificarea cererilor cât si descrierea planelor de acces la informatiile din baza de date. Fata de limbajele de cereri de tip SQL sau QUEL care permit introducerea cu usurinta a unor noi tipuri de date atomice si a unor functii (predicate) specifice acestora, limbajele algebrice faciliteaza si extinderea limbajului cu noi operatori relationali.

O algebra multisort este alcatuita dintr-o colectie de multimi (sorturi) si functii (dependente functionale) între aceste multimi. In SGBD-ul Gral un sort este folosit pentru modelarea relatiilor, celelalte corespund tipurilor de date atomice. Operatiile caracteristice pot fi relationale (de exemplu selectia sau filtrarea) sau operatii asociate tipurilor de date atomice (de exemplu operatii aritmetice sau operatii geometrice de felul intersectiei de poligoane). Sorturile si operatorii indepen-denti de aplicatie sunt grupate în asa numitul "cadru de modelare". Operatorii suportati sunt relationali binari (reuniune, produs cartezian, filtrare, concatenare), relationali unari (selec-tie, proiectie, ordonare, extend - care reali-zeaza extinderea unei relatii cu un atribut) operatorul extract (REL $\rightarrow$ ATOM) si ope-ratorii booleani (and, or, not). In portiunea bazei de date care este specifica unei aplicatii se pot descrie atât operatii referitoare la tipuri de date atomice cât si la relatii.

Modelul de date de aplicatie prezentat de Güting se numeste algebra geo-relationala si este adecvat tratarii aplicatiilor geometrice în plan fiind destinat unor aplicatii GIS. Sunt definite tipuri pentru puncte, linii poligonale si regiuni plane (fara goluri). Operatorii asociati tipurilor spatiale (POINT, LINE, PGON, AREA, REG={PGON,AREA}, EXT={LINE} $\cup$ REG, GEO={POINT} $\cup$ EXT) sunt:

inside (GEO x REG $\rightarrow$ BOOL),
dist (POINT x POINT $\rightarrow$ REAL),
length (LINE $\rightarrow$ REAL),
perimeter si area (REG $\rightarrow$ REAL).

Daca consideram o baza de date referitoare la orase, state, râuri si autostrazi, ale carei relatii sunt:

oras (onume:STR, centru:POINT, pop:INT),

judet(jnume:STR, regiune:AREA, pop:INT),
rau (mume:str, curs:LINE),
auto (anume:STR, drum:LINE).

Câteva exemple de cereri în acest limbaj algebric sunt:

oras, judet join [centru inside regiune]

project [onume, jnume]

Asociaza la fiecare oras numele judetului în care este situat respectivul oras.

judet select [jnume = "Prahova"] rau

join [curs intersect regiune]

project [mume]

Determina numele râurilor care trec prin judetul Prahova.

Extinderea limbajului de interogare prin specificarea de noi tipuri de date sau operatori necesita descrierea unor aspecte sintactice cum ar fi pozitia operatorilor fata de operanzi sau structura listei de parametri a unui operator. In general, daca cel putin unul din operanzi este o relatie se utilizeaza o notatie postfixata.

Intr-un SGBD extensibil este necesar ca proiectantul de aplicatii sa înțeleaga limbajul de specificare a planului de executie al cererilor; acesta poate fi bazat tot pe modelul algebrei multisort. In sistemul Gral, algebra asociata limbajului de cereri se zice descriptiva iar cea în care se specifica modul de executie al cererilor este algebra executabila. Pentru a specifica prelucrari eficiente la nivelul alge-brei executabile se pot descrie structurile de indexare, ordinea în care se fac operatiile asupra tuplelor relatiilor, astfel încât la fiecare acces la suportul de informatie externa sa fie prelucrata cât mai multa informatie.

Dintre sorturile specifice algebrei executabile pot fi mentionate:

REL - o multime de sorturi în care fiecare din elemente corespunde unei reprezentari fizice (index primar) a unei relatii. Valori posibile sunt SREL (reprezentare secventiala) STID-REL (o relatie specificata în acest fel poate fi accesata secvential sau printr-un index sec-vential) BTREE (reprezentare sub forma unui arbore B* ordonat dupa attributele relatiei).

INDEX - o multime de sorturi ale carei elemente sunt structuri de index secundar. Valorile posibile sunt SBTREE (arbore B* utilizat ca index secundar) SGRID (fisier grila) SMLGRID (fisier multigrila folosit pentru

indexarea într-o multime de drep-tunghiuri care pot delimita obiecte de forma unor linii poligonale).

TUPLE este un sort care descrie structura unui tuplu; ATOM - sort pentru descrierea de obiecte atomice ele sunt caracteristice si algebrei descriptive; TIDSEQ e un sort ce descrie o multime de identificatori de tupluri la care se poate avea acces printr-una din reprezentarile din REL.

Exemple de operatori ai algebrei executabile sunt:

scan:REL $\rightarrow$ SREL (parcurge toate tuplele unei relatii indiferent de modul de repre-zentare si verifica satisfacerea unei conditii)

sort:REL $\rightarrow$ SREL (implementeaza opera-torul ord din algebra descriptiva si permite filtrarea tuplelor obtinute prin sortare în conformitate cu valoarea unei expresii booleene de filtrare).

De exemplu cererea "determina tuplul din relatia orase care corespunde orasului Ploiesti":

oras select [onume = "Ploiesti"]

are corespondentul în algebra executabila:

oras "Ploiesti" exactmatch [true; tuple]

Extensibilitatea SGBD-ului Gral poate fi deci realizata la trei nivele: extensibilitate prin adaugare de noi tipuri de date, extensibilitate prin adaugare de noi operatori executabili si extensibilitatea prin adaugarea unor operatori descriptivi si a regulilor de optimizare asociate.

Bibliografie

[Catt 94] R. G. Cattell, Object Data Management, Addison Wesley Publishing Company, 1994.

[Egen 94] M. J. Egenhoffer, Spatial SQL: Q Query and Presentation Language, IEEE Transactions on Knowledge and Data Engineering, February 1994.

[Gupta 91] A. Gupta, T. Weymouth, R. Jain, Semantic Queries with Pictures: The VIMSYS Model, Proceedings of the 17-th International Conference on VLDB, 1991.

[Gütt 89] R. H. Gütting, Gral: An Extensible Relational Database System for Geometric Applications, Proceedings of the Fifteenth International Conference on Very Large Databases, 1989.

[Khos 96] J. Khoshafian, A. B. Baker,

Multimedia and Imaging Databases, Morgan Kauffman Publishers Inc., 1996.

[OMG 92] OMG ORBTF, Common Object Request Broker Architecture, Object Management Group, Framingham Massachussets, 1992.

[Samet 94] H. Samet, Spatial Data Models and Query Processing, aparut în Modern Database Systems, W. Kim (ed.), Addison Wesley, Reading, MA, 1994.