



Calculul relațional



Algebra relațională este un limbaj procedural - expresiile specifică pas cu pas construirea rezultatului

Calculul relațional - familie de limbaje de interogare, bazate pe calculul cu predicate de ordinul întâi (limbaje descriptive)

- interogarea este specificată prin proprietățile rezultatului
- contează mai puțin procedura ce trebuie urmată

Versiuni ale calculului relațional:

- Calculul relațional *pe domenii* - prezintă caracteristicile de bază ale acestor limbaje
- Calculul relațional *pe tupluri* - formează bazele multor constructori disponibili pentru interogarea în SQL

Calculul relațional pe domenii

Expresiile din calculul relațional pe domenii au forma

$$\{A_1 : x_1, \dots, A_k : x_k \mid f\},$$

$A_1 \dots, A_k$ - attribute distincte

$x_1 \dots, x_k$ - variabile (pentru simplitate le vom considera distincte)

f - formulă construită pe baza următoarelor reguli:

$$\{NrInreg : m, Nume : n, Varsta : v, Salariu : s \mid ANGAJAT(m, n, v, s) \wedge s > 40\}$$

a) există două tipuri de formule atomice:

- $R(A_1 : x_1, \dots, A_p : x_p)$ unde $R(A_1, \dots, A_p)$ este o schemă relațională și

$x_1 \dots, x_p$ variabile

- $x\theta y$ sau $x\theta c$,

x și y variabile, c constantă

θ operator de comparație ($=, \neq, \leq, <, \geq, >$)

b) dacă f_1 și f_2 sunt formule, atunci $f_1 \vee f_2$, $f_1 \wedge f_2$ și $\neg f_1$ sunt formule

c) dacă f este o formulă și x variabilă ce apare în f atunci $\exists x(f)$ și $\forall x(f)$ sunt formule cu

\exists - cuantificatorul existențial

\forall - cuantificatorul universal

$\{NrInreg : m, Nume : n, Varsta : v, Salariu : s \mid ANGAJAT(m, n, v, s) \wedge s > 40\}$

Lista de perechi $A_1 : x_1, \dots, A_k : x_k$ se numește **listă țintă** deoarece definește structura rezultatului;

- rezultatul va consta din relația definită pe $A_1 \dots, A_k$ care conține tupluri ale căror valori, substituite cu $x_1 \dots, x_k$ asigură că formula f este adevărată

Valoarea de adevăr a unei formule f se stabilește pe baza următoarelor reguli:

- formula atomică $R(A_1 : x_1, \dots, A_p : x_p)$ este adevărată pentru acele valori ale lui x_1, \dots, x_p ce formează un tuplu în R
- formula atomică $x\theta y$ este adevărată pentru acele valori ale lui x și y astfel încât valoarea lui x se află în relația θ cu valoarea lui y
 - în același mod se definește valoarea de adevăr a formulei $x\theta c$
- semnificația conectorilor \vee, \wedge, \neg este cea uzuală
- $\exists x(f)$ este adevărată dacă există cel puțin o valoare a lui x pentru care formula f este adevărată
- $\forall x(f)$ este adevărată dacă formula f este adevărată pentru toate valorile posibile ale lui x

Modul de exprimare a interogărilor în calculul pe domenii

Exemple

1. Se consideră baza de date cu schema:

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

Să se găsească numerele de înregistrare, numele, vârsta și salariul pentru toți angajații ce câștigă mai mult de 40

În algebra relațională: $\sigma_{Salariu > 40}(ANGAJAT)$

În calculul relațional pe domenii ?

Modul de exprimare a interogărilor în calculul pe domenii

Exemple

1. Să se găsească numerele de înregistrare, numele, vârsta și salariul pentru toți angajații ce câștigă mai mult de 40

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

În calculul relațional pe domenii:

$\{NrInreg : m, Nume : n, Varsta : v, Salariu : s \mid ANGAJAT(m, n, v, s) \wedge s > 40\}$

2. Să se găsească numerele de înregistrare pentru supervizorii acelor angajați ce câștigă mai mult de 40

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

În algebra relațională:

$$\pi_{NrSup}(SUPERVIZOR \bowtie_{NrAng=NrInreg} (\sigma_{Salariu>40}(ANGAJAT)))$$

În calculul relațional pe domenii ?

2. Să se găsească numerele de înregistrare pentru supervizorii acelor angajați ce câștigă mai mult de 40

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

În calculul relațional pe domenii:

$$\{NrSup : n \mid ANGAJAT(NrInreg : m, Nume : u, Varsta : v, Salariu : s) \wedge s > 40 \wedge SUPERVIZOR(NrSup : n, NrAng : m)\}$$

Variabila **m**, comună ambelor condiții atomice, realizează corespondența între tuplurile specificate în joncțiune.

Se pot utiliza cuantificatori existențiali pentru toate variabilele ce nu apar în lista țintă \Rightarrow se complică formularea cererii.

Calculul relațional pe domenii

Dacă într-o expresie este necesară implicarea unor tupluri diferite ale aceleiași relații este suficientă impunerea mai multor condiții asupra aceluiasi predicat din formulă, utilizând variabile diferite.

3. Să se găsească numele și salariile pentru supervizorii acelor angajați ce câștigă mai mult de 40

În algebra relațională:

$$\pi_{Nume, Salariu} (ANGAJAT \bowtie_{NrInreg=NrSup} (SUPERVIZOR \bowtie_{NrAng=NrInreg} (\sigma_{Salariu \geq 40} (ANGAJAT))))$$

În calculul relațional pe domenii ?

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

3. Să se găsească numele și salariile pentru supervizorii acelor angajați ce câștigă mai mult de 40

ANGAJAT (NrInreg, Nume, Varsta, Salariu)

SUPERVIZOR (NrSup, NrAng)

În calculul relațional pe domenii - pentru fiecare tuplu din rezultat trebuie să existe trei tupluri: un tuplu corespunde unui angajat ce câștigă mai mult de 40, al doilea tuplu indică supervizorul său și ultimul furnizează informații detaliate despre supervizor:

$$\{ \text{Nume} : ns, \text{Salariu} : ss \mid \\ \text{ANGAJAT}(\text{NrInreg} : ni, \text{Nume} : u, \text{Varsta} : v, \text{Salariu} : s) \wedge s > 40 \wedge \\ \text{SUPERVIZOR}(\text{NrSup} : n, \text{NrAng} : ni) \wedge \\ \text{ANGAJAT}(\text{NrInreg} : n, \text{Nume} : ns, \text{Varsta} : q, \text{Salariu} : ss) \}$$

Dezavantaje ale calculului relațional pe domenii

Limitări ce au importanță din punct de vedere practic:

$$\{A_1 : x_1, A_2 : x_2 \mid R(A_1 : x_1) \wedge x_2 = x_1\}$$

Expresia

produce ca rezultat ???

- dacă domeniul se schimbă?
- dacă domeniul este infinit?

Dezavantaje ale calculului relațional pe domenii

Limitări ce au importanță din punct de vedere practic:

Expresia $\{A_1 : x_1, A_2 : x_2 \mid R(A_1 : x_1) \wedge x_2 = x_2\}$ produce ca rezultat o relație pe A_1 și A_2 formată din tupluri ale căror valori pe A_1 apar în relația R , și valoarea pe A_2 poate fi orice valoare din domeniu (deoarece condiția este întotdeauna adevărată)

- dacă domeniul se schimbă, spre exemplu de la întreg între 0 și 99 la întreg între 0 și 999, răspunsul interogării se schimbă de asemenea;
- dacă **domeniul este infinit**, atunci și **rezultatul este infinit**, ceea ce este de neacceptat.

O observație similară poate fi făcută și referitor la expresia

$$\{A_1 : x_1 \mid \neg R(A_1 : x_1)\}$$

al cărei rezultat este o relație ce conține acele valori din domeniul A_1 care nu apar în relația R .

Definiții

1. O expresie a unui limbaj de interogare este independentă de domeniu dacă rezultatul său în fiecare instanță a bazei de date nu variază dacă se modifică domeniul pe care se evaluează expresia.
2. Un limbaj este independent de domeniu dacă toate expresiile lui sunt independente de domeniu.

Pe baza acestor definiții putem afirma că:

- algebra relațională este independentă de domeniu (construind rezultatul pe baza relațiilor din baza de date, fără a face referire la domeniile atributelor);
- calculul relațional pe domenii este dependent de domeniu.

3. Spunem ca **două limbaje de programare sunt echivalente** dacă pentru orice expresie dintr-un limbaj există o expresie echivalentă în cel de-al doilea limbaj și invers.
 - **Algebra relațională și calculul pe domenii nu sunt echivalente**, deoarece calculul pe domenii permite formularea unor expresii ce sunt dependente de domeniu
 - O echivalență se poate obține dacă limităm calculul pe domenii la o submulțime de expresii independente de domeniu

Un alt **dezavantaj** al calculului relațional pe domenii este dat de **numărul mare de variabile cerute într-o expresie**, de obicei o variabilă pentru fiecare atribut

Singurul limbaj bazat, într-o anumită măsură, pe calculul pe domenii este QBE (Query by Example)

Calculul relațional pe tupluri

Expresiile din calculul pe tupluri au forma $\{T \mid L \mid f\}$

$$\{s.NrSup \mid e(ANGAJAT), s(SUPERVIZOR) \mid \\ e.NrInreg = s.NrAng \wedge e.Salariu > 40\}$$

- T - lista țintă, compusă din elemente de tipul $Y:x.Z$ (sau mai simplu $x.Z$ ca abreviere pentru $Z:x.Z$), cu x variabilă și Y, Z secvențe de atribute de lungimi egale
 - atributele din Z trebuie să apară în schema relației ce constituie gama de valori a lui x
 - vom scrie $x.*$ ca abreviere pentru $X:x.X$, unde gama de valori a variabilei x este o relație definită pe atributele X
- L - lista gamei de valori, în care sunt enumerate variabilele libere din formula f , împreună cu gamele de variație corespunzătoare
 - L este o listă de elemente de tipul $x(R)$, cu x variabilă și R relație
 - dacă $x(R)$ este în lista gamei de valori, atunci x reprezintă tuplurile din R

■ f - formulă constituită din:

- atomi de tipul $x.A\theta c$ sau $x_1.A_1\theta x_2.A_2$ care compară valoarea lui x corespunzătoare atributului A cu constanta c și valoarea lui x_1 pe atributul A_1 cu cea a lui x_2 pe atributul A_2
- conectori (\wedge, \vee, \neg)
- cuantificatori:

$\exists x(R)(f)$ - există un tuplu x în R ce satisface f

$\forall x(R)(f)$ - orice tuplu x din R satisface f

Notă. În timp ce se introduce variabila x , o declarație a gamei de valori de forma $x(R)$ specifică faptul că x poate lua ca valori numai tupluri din relația $R \Rightarrow$ acest limbaj nu necesită formularea unor condiții atomice ca în calculul pe domenii care să indice că un tuplu aparține unei anumite relații.

Exemple. Se consideră baza de date cu schema:

ANGAJAT (NrInreg, Nume, Varsta, Salariu)
SUPERVIZOR (NrSup, NrAng)

- 1) Să se găsească numerele de înregistrare, numele, vârsta și salariul pentru toți angajații ce câștigă mai mult de 40

$$\{e.* \mid e(ANGAJAT) \mid e.Salariu > 40\}$$

Exemple. Se consideră baza de date cu schema:

ANGAJAT (NrInreg, Nume, Varsta, Salariu)
SUPERVIZOR (NrSup, NrAng)

- 2) Să se găsească numerele de înregistrare pentru supervizorii acelor angajați ce câștigă mai mult de 40

$$\{s.NrSup \mid e(ANGAJAT), s(SUPERVIZOR) \mid \\ e.NrInreg = s.NrAng \wedge e.Salariu > 40\}$$

Exemple. Se consideră baza de date cu schema:

ANGAJAT (NrInreg, Nume, Varsta, Salariu)
SUPERVIZOR (NrSup, NrAng)

3) Să se găsească numele și salariile pentru supervizorii acelor angajați ce câștigă mai mult de 40

$$\{NumeS, SalariuS : e'.(Nume, Salariu) | \\ e'(ANGAJAT), e(ANGAJAT), s(SUPERVIZOR) | \\ e'.NrInreg = s.NrSup \wedge s.NrAng = e.NrInreg \wedge e.Salariu > 40\}$$

Observații

- Nu orice interogare din algebra relațională sau din calculul relațional pe domenii se poate exprima în calculul relațional pe tupluri

Exemplu - interogările care în algebra relațională necesită operatorul reuniune nu pot fi exprimate prin calculul pe tupluri

- SQL pune la dispoziție posibilitatea construirii în mod explicit a reuniunii, deoarece aspectele declarative ale SQL se bazează pe calculul pe tupluri cu declarația gamei de valori



Probleme propuse

1. Se consideră baza de date alcătuită din relațiile următoare:

FILME (NumarFilm, Titlu, Regizor, An, CostProductie)

ARTISTI (NumarActor, Nume, Prenume, Sex, DataNastere, Nationalitate)

ROLURI (NumarFilm, NumarActor, Personaj)

Exprimați următoarele interogări în calculul relațional pe domenii și în calculul relațional pe tupluri:

- a) Să se găsească titlurile filmelor în care joacă Henry Fonda
- b) Să se găsească titlurile filmelor pentru care regizorul este și actor
- c) Să se găsească actorii care au interpretat două personaje în același film; precizați titlul filmului, numele și prenumele actorului și numele personajelor interpretate
- d) Să se găsească titlurile filmelor în care actorii sunt toți de același sex



2. Se consideră baza de date care conține următoarele relații:

CURSURI (Numar, Facultate, TitluCurs, Titular)

STUDENTI (Numar, Nume, Prenume, Facultate)

TITULARI (Numar, Nume, Prenume)

EXAMENE (Student, Curs, Nota, Data)

PLAN_INVATAMANT (Student, Curs, An)

Formulați în calculul relațional pe domenii și în calculul relațional pe tupluri interogările care produc:

- a) studenții care au obținut nota 10 la cel puțin un examen, precizând pentru fiecare numele, prenumele și data când au obținut prima notă de 10
- b) pentru fiecare curs de la Automatică, studenții care au trecut examenul în ultima sesiune
- c) studenții care au trecut toate examenele cerute de planul de învățământ
- d) pentru fiecare curs de la Electronică, studenții care au obținut cea mai mare notă
- e) numele și prenumele studenților care au luat examenul la un curs al cărui titular are același nume ca și studentul



3. Pentru baza de date cu relațiile:

ORASE (Nume, Judet, NumarLocuitori)

TRAVERSARI (Oras, Râu)

RÂURI (Râu, Lungime),

formulați următoarele interogări în calculul relațional pe domenii și în calculul relațional pe tupluri :

- a) găsiți numele, județele și numărul de locuitori pentru orașele care au mai mult de 50000 de locuitori și sunt traversate de Siret sau Mureș
- b) găsiți orașele care sunt traversate de cel puțin două râuri, precizând numele orașului și numele celui mai lung râu care-l traversează