

Marin Fotache

Fiola de SQL (6)

Metafizică și SQL. Theta-joncțiunea

Nu cred că poate fi numit profesor adevărat cel care nu-și consternează, baremi o dată pe semestru, studenții. Dacă nici măcar unul dintre cursuri nu se soldează cu ceva figuri contrariate, zăpăcite (pe lângă cele adormite), ce nu-și pot masca o expresie (a feței) de genul “ce naiba vrea să spună?”, este evident că nu faci știință și ești un cadru didactic de mâna n, unde $n \geq 2$.

Eu, unul, am încercat de multe ori să-mi salvez imaginea de curs (și, normal, de profesor) serios/profund prin prezentarea theta-joncțiunii și a diviziunii relaționale. Este drept, theta-joncțiunea și diviziunea sunt legați nu neapărat de SQL, ci de algebra relațională. Cum însă SQL este fundamentat teoretic de algebra relațională, nu văd de ce nu i-am putea pune în cârcă, cu șalanță sau nonșalanță, acești operatori.

Joncțiunea reprezintă fuziunea a două relații care au o proprietate comună. Fie două relații, notate: $R1 (A1, A2, \dots, An)$ și $R2 (B1, B2, \dots, Bp)$. Fie Ai și Bj două atribute definite pe același domeniu, și θ ansamblul operatorilor pentru comparații $\{=, >, \geq, <, \leq, \neq\}$, ce pot fi aplicați celor două atribute Ai și Bj . Joncțiunea relației $R1$, prin Ai , cu relația $R2$, prin Bj , notată $R1 (Ai \theta Bj) R2$ sau $R1 \otimes_{\theta} R2$, este relația ale cărei tupluri sunt obținute prin concatenarea fiecărui tuplu al relației $R1$ cu tuplurile relației $R2$, pentru care este verificată condiția θ instituită între Ai și Bj . Se notează:

$R1(Ai \otimes Bj)R2 = \{t \mid t \in R1 \otimes R2 \text{ și } t(Ai) \theta t(Bj)\}$

Pentru doritorii de senzații tari, în algebra relațională există o notație și mai pronunțată matematizată, ce utilizează simbolurile alfabeticului grecesc, dar, de data aceasta, vreau să-mi menajez editorul de ecuații al Office-ului.

Joncțiunea este echivalența unui produs cartezian urmat de o selecție. Joncțiunea definită mai sus este referită în lucrările de specialitate ca *theta-joncțiune*. În lucrul cu BDR se utilizează cu precădere *echi-joncțiunea*, ce reprezintă un caz particular al theta-joncțiunii, atunci când θ este operatorul de egalitate (“=”). În *figura alăturată* este prezentat mecanismul în doi pași de derulare a theta-joncțiunii:

$R \leftarrow \text{JONCTIUNE} (R1, R2; R1.A \geq R2.E)$

Cazurile practice de utilizare a theta-joncțiunii nu sunt chiar atât de frecvente. Mărturisesc că mi-a luat ceva timp până să găsesc

niște exemple rezonabile care să arate pe deplin utilitatea acestui operator (acum pare simplu...).

Începem cu un exemplu bazat pe două tabele de care ne-am folosit repetat pe parcursul fiolilor de până acum: PERSONAL{Marca, NumePren, DataNast, Compart, MarcaSef, SalTarifar} și SPORURI {An, Luna, Marca, SporVechime, SporNoapte, SporCD, AlteSpor}.

Care sunt angajații cu salariul tarifar mai mare decât al salariatului cu numele ANGAJAT 4?

Soluția 1, valabilă deopotrivă pentru VFP și Oracle, utilizează o subconsultare:

```
SELECT *
FROM PERSONAL
WHERE SalTarifar >
  (SELECT SalTarifar
   FROM PERSONAL
   WHERE NumePren='ANGAJAT 4')
```

Soluția 2 este însă bazată pe theta-joncțiune. Tabelele theta-joncționate sunt două instanțe ale PERSONAL. Predicatul joncțiunii este $P1.SalTarifar > P2.SalTarifar$ AND $P2.NumePren = 'ANGAJAT 4'$:

```
SELECT P1.*
FROM PERSONAL P1, PERSONAL P2
WHERE P1.SalTarifar >
  P2.SalTarifar
AND P2.NumePren = 'ANGAJAT 4'
```

După “calculul” produsului cartezian al $P1$ și $P2$, sunt extrase liniile acestuia care verifică condiția formulată prin predicatul de selecție.

Acolo unde este posibil (DB2, Oracle etc.), se poate recurge și la o variantă în care predicatul theta-joncțiunii este mai “pur”, prin faptul că $P2$ conține numai linia corespunzătoare ANGAJATULUI 4:

Soluția 3 - Oracle

```
SELECT P1.*
FROM PERSONAL2 P1,
(SELECT * FROM PERSONAL2
 WHERE NumePren = 'ANGAJAT 4') P2
WHERE P1.SalTarifar > P2.SalTarifar
```

Dintre cele două variante, *subconsultări* vs. *theta-joncțiune*, prima este de preferat, datorită unui consum mai mic de resurse (joncțiunea externă presupune produsul cartezian).

Există însă situații în care theta-joncțiunea constituie o soluție directă și elegantă pentru redactarea unei interogări. Pentru exemplificare trecem într-o cu totul altă zonă profesională. Acum câțiva ani am avut un contract pe vreo două sezoane cu Facultatea de Bioinginerie Medicală. Deoarece: se apropia acreditarea (facultății), aveau prea mulți strănieri în lot, eu am refuzat naturalizarea “bioingineriasco-medicală”, păstrându-mi-o pe cea “științio-economică”, colaborarea s-a întrerupt brusc și aproape amiabil.

Dincolo de amintirile plăcute, am rămas și cu un exemplu pe cînte pentru theta-joncțiune. Într-un spital, să-l considerăm de urgență, interesează evidența internărilor și a doctorilor de gardă. Ne limităm la trei tabele: PACIENȚI - ce conține date generale despre pacienții internați: codul numeric personal (CNPPacient), numele și prenumele, data nașterii, adresa, localitatea și județul de domiciliu; GĂRZI, în care se evidențiază: data și ora de început a gărzii (D0Inițială), data și ora de sfârșit a

Mecanismul de (theta)joncționare

R1			R ← R1 ⋈ R2					
A	B	C	A	B	R1.C	R2.C	D	E
20	XYZ	30	20	XYZ	30	25	XYZ	30
30	XXZ	20	20	XYZ	30	40	YYX	25
40	YYX	25	20	XYZ	30	30	XXZ	40
			30	XXZ	20	25	XYZ	30
			30	XXZ	20	40	YYX	25
			30	XXZ	20	30	XXZ	40
			40	YYX	25	25	XYZ	30
			40	YYX	25	40	YYX	25
			40	YYX	25	30	XXZ	40

R = SELECTIE (R', A >= E)

A	B	R1.C	R2.C	D	E
30	XXZ	20	25	XYZ	30
30	XXZ	20	40	YYX	25
40	YYX	25	25	XYZ	30

gărzii (DOFinală) și numele medicului de gardă; în sfârșit, INTERNĂRI pentru care atributele se referă la: data și ora internării (DOInternare), codul numeric personal al pacientului internat, diagnosticul și starea din momentul internării.

În *listing 1* este prezentat scriptul Oracle de creare a celor trei tabele. În VFP atributele DOInitiala, DOFinala, DOInternare sunt de tip DATETIME.

Listing 1. Scriptul Oracle de creare a tabelor PACIENȚI, GĂRZI și INTERNĂRI

```
CREATE TABLE PACIENTI (
  CNPPacient CHAR(13),
  Nume VARCHAR2(20),
  Prenume VARCHAR2(20),
  DataNasterii DATE,
  Adresa VARCHAR2(50),
  Localitate VARCHAR2(20),
  Judet CHAR(2),
  PRIMARY KEY (CNPPacient) );
```

```
CREATE TABLE GARZI (
  DOInitiala DATE,
  DOFinala DATE,
  MedicGarda VARCHAR2 (20),
  PRIMARY KEY (DOInitiala, DOFinala));
```

```
CREATE TABLE INTERNARI (
  DOInternare DATE,
  CNPPacient CHAR(13) REFERENCES
  PACIENTI(CNPPacient),
  Diagnostic VARCHAR2(40),
  Stare VARCHAR2(30),
  PRIMARY KEY (DOInternare, CNPPacient));
```

Formulăm problema care ne conduce abrupt către theta-joncțiune:

Care este data și ora internării fiecărui pacient (identificat prin CNP), diagnosticul și starea din momentul internării, precum și numele doctorului care era de gardă ?

N-avem încotro. Cea mai simplă soluție este joncționarea tabelor INTERNĂRI și GĂRZI după condiția DOInternare >= DOInitiala AND DOInternare <= DOFinala. Pentru economie de caractere folosim operatorul BETWEEN:

Soluția VFP

```
SELECT DOInternare AS DataOraInternare, ;
  CNPPacient, Diagnostic, Stare, ;
  DOInitiala AS Inceput, ;
  DOFinala AS Sfirsit, MedicGarda ;
FROM INTERNARI, GARZI ;
```

```
WHERE INTERNARI.DOInternare BETWEEN ;
  GARZI.DOInitiala AND GARZI.DOFinala
```

Soluția Oracle

Oracle 8 nu face distincția dintre câmpurile de tip DATE (dată calendaristică) și cele de tip DATETIME (dată plus ora), primul incluzându-l pe cel de-al doilea. Pentru a vizualiza și ora (minutul și secunda) momentelor internării, de început și de sfârșit ale gărzii, trebuie utilizat un șablon în funcția TO_CHAR:

```
SELECT TO_CHAR(DOInternare,
  'DD-MM-YYYY HH24:MI:SS') AS DataOraInternare,
  CNPPacient, Diagnostic, Stare,
  TO_CHAR(DOInitiala,'DD-MM-YYYY HH24:MI:SS') AS
  Inceput, TO_CHAR(DOFinala, 'DD-MM-YYYY
  HH24:MI:SS') AS Sfirsit, MedicGarda
FROM INTERNARI, GARZI
WHERE INTERNARI.DOInternare BETWEEN
  GARZI.DOInitiala AND GARZI.DOFinala
```

Pot fi formulate și soluții care să ocolească theta-joncțiunea. Însă, după cunoștințele mele, varianta prezentată este cea mai la îndemână.

Marin Fotache, fotache@uaic.ro