Validitatea în normalizarea 3NF prin sinteza

Prof.dr.Valer ROSCA
Universitatea “Lucian Blaga”, Sibiu

Asigurarea corectitudinii datelor pe care le contine o baza de date este o problema de maxima importanta. In acest sens, pentru bazele de date relationale, este utila semantica datelor, specificata prin dependente functionale (DF). Dar, prezenta unor categorii de DF, aduce si neajunsuri în ceeace priveste actualizarea acestor baze (anomalii), de unde apare necesitatea unui proces de normalizare. Normalizarea amelioreaza structura bazei de date, elimin{nd DF care creaza probleme si aduce aceasta structura la o anumita stare, denumita forma normala. Forma normala, denumita 3NF, este o forma acceptabila, în majoritatea cazurilor si ea face obiectul analizei care urmeaza.

Cuvinte cheie: *normalizare, dependenta functionala, forma normala.*

1. Scop

Normalizarea prin sinteza este o metoda care pleaca de la multimea atributelor rela-tiei globale si o multime de DF-uri, puse în evidenta în procesul de analiza si cons-truieste relatiile elementare ale unei des-compuneri, selectând atributele acestora în-tr-un anumit mod. În anumite conditii, sin-teza poate asigura o descompunere valida si care pastreaza dependentele functionale.

2. Preliminarii

Pentru fixarea notatiilor si a sensului noti-unilor utilizate, se face o succinta prezen-tare a partii structurale a modelului relatio-nal, relevanta pentru scopul propus.

D1: Fie $D=\{D_1, D_2, ..., D_n\}$ o familie de domenii si $A=\{A_1, A_2, \dots, A_n\}$ o familie de atribute, cu A_i definit pe D_i. Se spune ca:

- cuplul S=@A, D# este o **schema de relatie** n-ara (**intensie**);
- o submultime $E \subseteq D_1 x D_2 x ... x D_n$ este o **extensie** (instantiere) a lui S;
- $(a_1, a_2, ..., a_n)$ este un **tuplu** al extensiei E.

Domeniile si atributele sunt atomice, adica valorile lor sunt nedecompozabile. Pentru simplitate, familia de domenii se omite si o submultime de atribute X={A,..,B} se desemneaza ca atribut compus si se noteaza în forma X=(A,B). Acest mod de abordare impune o anumita ordine a atributelor, ceea ce se va presupune în continuare. Daca $t \in E$, atunci valorile lui X pentru t cons-tituie un tuplu notat t(X) care se obtine ca proiectie dupa atributele simple din care X este constituit.

D2: Un cuplu de atribute @X,Y#, $Y \not\subset X$ se spune ca defineste o **dependenta functionala** (DF) si se noteaza $X \rightarrow Y$, daca pentru orice extensie E si oricare doua tuple t_1, $t_2 \in E$, din $t_1(X)=t_2(X)$ rezulta $t_1(Y)=t_2(Y)$. Despre o extensie E se spune ca **verfica dependenta functionala**, daca pentru ea sunt adevarate relatiile din defi-nitia anterioara.

Fie F o multime de DF si f o DF care nu apartine lui F. Se spune ca f este **implicata** (derivata) de F, daca orice extensie E a lui R care verifica dependentele lui F, verifica si dependenta data. Multimea F^+, a dependentelor implicate de F, este denumita **închidere** a lui F. Exista algoritmi pentru determinarea închiderii lui F.

Doua multimi de DF sunt **echivalente**, da-ca ele au aceeasi închidere. Daca, prin eliminarea dependentei f din F, se obtine aceeasi închidere pentru F, atunci se spune ca f este **redundanta** sau **superflua**.

D3: Un cuplu <S, F>, în care S este o schema de relatie n-ara, iar F o multime de DF pe atributele lui S, se numeste **schema relationala** si se noteaza R=<S, F> sau R(S), daca nu este necesar sa se puna în evidenta F.

Un atribut simplu sau compus K se spune ca este **cheie** a lui R, daca exista DF $K \rightarrow S$ si K este multime minimala (se poa-te arata ca $K \rightarrow S$ este echivalent cu $K \rightarrow A_1$, …, $K \rightarrow A_v$).

D4: Fie R=<S, F>, Y, Z atribute ale lui R. Se spune ca Y **depinde partial** de Z, daca exista $X \subset Z$ si $X \rightarrow Y$, altfel Y **depinde total** de Z. Y **depinde tranzitiv** de Z, daca exista X, $Y \rightarrow X$ si $Y \not\subset X$, astfel încât $Z \rightarrow X$ si $X \rightarrow Y$.

Dependentele partiale si tranzitive mode-leaza legaturi nedorite între atribute care impieteaza asupra schemei relationale, de-oarece produc anomalii de actualizare. În raport cu ele, se definesc formele normale care intereseaza aici.

D5: Fiind data o schema relationala R=<S, F>, o cheie K a lui R si X, Y atri-bute neprimitive, se spune ca R este în:

- **prima forma normala** (1NF), daca ori-care ar fi X exista $K \rightarrow X$;
- **a doua forma normala** (2NF), daca R este în 1NF si orice X depinde total K;
- **a treia forma normala** (3NF), daca R este în 2NF si orice X depinde netranzitiv de K.

Formele normale amelioreaza succesiv schema relationala R, din punctul de vedere al anomaliilor de actualizare, astfel încât forma 3NF este considerata acceptabila pentru multe cazuri de baze de date.

D6: Fie R=<S, F> o schema relationala. Se spune ca R_1=<S_1, F_1> este o **subschema relationala** a lui R, daca $S_1 \subset S$ si F_1 este multimea de DF $X \rightarrow Y$ din F cu $X \subset S_1$ si $Y \subset S_1$.

D7: Fiind data schema relationala R=<S, F>, o multime $\Delta = \{R_1, R_2, ..., R_p\}$ de subscheme ale lui R, cu R_i=<S_i, F_i>, este o **descompunere** a lui R, daca $S = \cup S_i$.

Pentru uniformitate, subschemele sunt de-numite scheme relationale, iar R este numi-ta schema globala. Intereseaza descompu-neri ale lui R corecte, din anumite puncte de vedere, asa cum rezulta din definitia care urmeaza.

D8: Fie F o multime de DF si Δ o des-compunere a lui R. Se spune ca descom-punerea:

- **pastreaza continutul** bazei (este **valida)** în raport cu F, daca pentru orice extensie E a lui R care verifica pe F, E este **join natural** al proiectiilor sale dupa Δ, adica E=JOIN(E(R_1),..,E(Rp)), unde cu E(R_i) s-a notat proiectia lui E dupa atributele lui R_i ;
- **pastreaza (conserva) dependentele functionale**, daca $F^+=(\cup F_i)^+$.

Descompunerea valida asigura reconstitui-rea unica, în orice moment, a relatiei glo-bale din din proiectiile acesteia dupa rela-tiile de descompunere.

3. Algoritmul de sinteza

Se considera multimea de DF ale lui F care au membrul drept atribut simplu, nu sunt partiale si nu sunt redundante: multimea astfel obtinuta, notata F^{min}, este echivalenta cu F si este denumita **acoperirea mini-mala** a lui F. Acoperirea minimala a lui F nu este unica. Exista algoritmi care permit determinarea directa a unei acoperiri mini-male.

Algoritmul de sinteza utilizeaza o acope-rire minimala a multimii F. Analiza nu pierde din generalitate, daca se presupune ca din acoperirea minimala se elimina de-pendentele de forma $Y \rightarrow X$, atunci când exista si dependenta $X \rightarrow Y$, deoarece pre-zenta lor în algoritm asigura numai redu-cerea numarului de relatii ale descom-punerii sintetizate. Algoritmul poate fi re-dat, în ceea ce este esential prin pasii care urmeaza.

Algoritmul 1:
1) - Se determina o acoperire minimal F^{min} a lui F;
2) - Se împarte acoperirea minimala în gru-puri de dependente: un grup F_i este format din toate dependentele acoperirii minimale care au acelasi membru stâng;
3) - Daca n#0 este numarul de grupuri construite în pasul anterior, se defineste descompunerea lui R ca fiind formata din relatii R_i=@S_i, F_i# i=1...n, unde S_i este multimea atributelor pe care le contin dependentele din F_i.

Algoritmul produce relatii care sunt, cel ptin, în forma normala 3NF. Într-adevar, în nici o relatie nu exista dependente partiale, din modul de constructie al acoperirii mini-male. Deasemenea, datorita modului de grupare, nu exista dependente tranzitive, deoarece, pentru oricare doua dependente $X \rightarrow Y$, $X \rightarrow Z$ dintr-o relatie, nu poate exista, în aceeasi relatie si dependenta $Z \rightarrow Y$.

4. Validitatea descompunerii

In ceeace priveste validitatea descompu-nerii sintetizate, se pot aduce exemple din care sa rezulte ca algoritmul nu o poate asigura în toate cazurile [4].
Daca se da o descompunere, realizata pe baza unei multimi F de DF, exista un **algoritm de verificare a validitatii** [5] ca-re se bazeaza pe construirea unei matrice V, cu p linii - câte una pentru fiecare relatie din descompunere - si n coloane, c{te una pentru fiecare atribut al relatiei globale. Algoritmul poate fi redat prin pasii care urmeaza.

Algoritmul 2:
a)- Se initializeaza matricea V:
$V_{ij}=a_i$ daca atributul A_i apartine relatiei R_i;
$V_{ij} = b_{ij}$ altminteri.
b)- Se transforma elementele matricei V, pe rând, pentru fiecare dependenta din F. Daca $X \rightarrow Y$ este o astfel de dependenta, atunci, pentru fiecare pereche de lini i,k (i@k) în care valorile corespunzatoare in-tersectiei cu coloanele atributelor din X co-incid, se pune aceeasi valoare, la intersectia cu coloana j a atributului Y, daca o astfel de valoare nu exista deja. Daca una din vechile valori este a_i, atunci aceasta este luata ca valoare comuna. Daca vechile va-lori sunt b_{ij} si b_{kj} , atunci b_{ij} este valoarea comuna.
c)- Daca nu s-au epuizat dependentele, se reia algoritmul de la pasul b.

Se poate demonstra teorema care urmeaza.
Teorema 1:
Algoritmul de verificare a validitatii este corect si descompunerea este valida daca, la terminarea sa, matricea V contine o linie de forma: a_1, a_2, ..., a_n.
Se poate, acum, formula o conditie sufi-cienta pentru ca o descompunere sintetizata sa fie valida.

Teorema 2:
Daca descompunerea sintetizata prin algo-ritmul 1 pastreaza dependentele functio-nale si cheia relatiei globale este continuta într-o relatie a acesteia, atunci descompu-nerea este valida.
Demonstratie:
Se aplica algoritmul de verficare al vali-ditatii. Pentru fiecare atribut A_j care figu-reaza în R_i, se pune a_j în coloana j si pe linia i (unde figureaza relatia R_i). Se com-pleteaza matricea V cu valori b_{ij}, daca A_j nu se afla în

relatia R_i. Pentru fiecare dependenta $A_j \rightarrow Y$ din F^{min} care figureaza într-un F_i, se aplica apoi pasii de trans-formare ai algoritmului de verificare. Daca K este cheia relatiei universale si ea apare într-o relatie R_i , deoarece descompunerea pastreaza dependentele functionale, atunci, în matricea transformata, apare a_j pe linia i pentru toate atributele A_j din schema glo-bala. Deci, descompunerea este valida.

Teorema 2 permite ameliorarea algorit-mului, în sensul adaugarii unei relatii care este formata numai de atributele cheii uni-versale si care nu are nici o dependenta functionala, atunci când nici una din rela-tiile descompunerii sintetizate nu contine pe K. Se poate formula corolarul care ur-meaza.

Corolarul 1:

Daca Δ este o descompunere a lui R, sin-tetizata prin algoritmul 1 care pastreaza de-pendentele functionale, atunci $\Delta \cup \{K, \emptyset\}$ este o descompunere valida a lui R.

Se poate formula o reciproca a teoremei 2.

Teorema 3:

Daca o descompunere a relatiei universale R este valida în raport cu F, atunci cel pu-tin o relatie a descompunerii contine cheia K a relatiei universale.

Demonstratie:

Se poate da o demonstratie prin reducere la absurd. Presupunem ca exista o descompu-nere valida a lui R si nici una din relatiile acesteia nu contine pe K.

Se aplica algoritmul de verificare a vali-ditatii si, deoarece descompunerea este va-lida, în matricea V va apare o linie i numai cu valori a_j. Aplicând inductia dupa etapele algoritmului de verificare, se deduce ca pentru oricare element a_j din linia i, depen-denta $R_i \rightarrow A_j$ apartine lui F^+. Cum, pe linia i apare câte o valoare a_j, pentru fiecare atribut al lui R, rezulta ca R_i este cheie a relatiei universale. Contradictia aparuta încheie demonstratia.

5. Concluzie

Din cele de mai sus, se poate trage con-cluzia ca algoritmii de normalizare prin sinteza, eventual ameliorati pe baza Co-rolarului 1, pot asigura construirea auto-matizata de descompuneri 3NF care sa fie valide în raport cu o multime de depen-dente functionale.

Bibliografie

1. Bâsca O. - Baze de date. Editura ALL, Bucuresti 1997.

2. Date C. J - An Introduction to Database Systems, Addison Wesley Publishing Co, 1990.

3. Lungu I, s.a - Baze de date: organizare, proiectare si implementare, Colectia Informatica, Grupul ALL, 1995.