

Cursul 8

Proiectarea bazelor de date - Partea 2 -

F. Radulescu, Curs: Baze de date

1

Sumar

1. Problematica. Anomalii.
2. Dependente functionale
3. Forme normale: FN1, FN2, FN3, FNBC
4. Descompunerea schemelor de relatii
5. Dependente multivalorice. Forma normala 5.

F. Radulescu, Curs: Baze de date I

2

FORME NORMALE

- ◆ Exista cateva seturi de conditii care ne arata ca o schema de relatie este corect proiectata in sensul ca ea nu permite aparitia anomaliilor prezentate la inceputul capitolului.
- ◆ Daca schema indeplineste cerintele unui anumit set de conditii se spune ca este in **forma normala** asociata acelui set.
- ◆ In continuare sunt prezentate formele normale Boyce-Codd si forma normala 3. In finalul acestui capitol va fi prezentata si forma normala 4 care se defineste in functie de alt tip de dependente, si anume dependentele multivalorice.

F. Radulescu, Curs: Baze de date

3

FNBC - DEFINITIE

Definitie. Fie R o schema de relatie si F multimea de dependente functionale asociata. Se spune ca R este in forma normala Boyce-Codd (FNBC) daca si numai daca oricare ar fi o dependenta netriviala $X \rightarrow Y$ din F atunci X este supercheie pentru R

F. Radulescu, Curs: Baze de date

4

FNBC – cont.

- ◆ Rezulta ca o schema de relatie este in FNBC daca si numai daca fiecare dependenta netriviala din F are in partea stanga o supercheie.
- ◆ Nu este necesar ca F sa fie in forma canonica dar nu se iau in considerare dependentele triviale (obtinute din prima axioma - de reflexivitate, de tipul $AB \rightarrow A$ sau $AB \rightarrow AB$).
- ◆ Observatie importanta: o schema fara nici o dependenta netriviala este in FNBC!

F. Radulescu, Curs: Baze de date

5

EXEMPLUL 1

- ◆ Relatia R = ABCDE avand
 $F = \{ A \rightarrow B, B \rightarrow A, A \rightarrow C, D \rightarrow E \}$
- ◆ Nu este in forma normala Boyce-Codd deoarece are cheile AD si BD dar nici o dependenta nu are in partea stanga o supercheie a lui R

F. Radulescu, Curs: Baze de date

6

EXEMPLUL 2

- ◆ Relatia $\text{Produce} = \text{IdP, NumeP, Qty, IdF}$ avand asociata multimea de dependente functionale
- ◆ $F_{\text{PRODUCE}} = \pi_{\text{PRODUCE}}(F) = \{ \text{IdP} \rightarrow \text{NumeP}, \text{IdP} \rightarrow \text{Qty}, \text{IdP} \rightarrow \text{IdF} \}$
- ◆ Este in forma normala Boyce-Codd: cheia unica a relatiei este IdP si toate dependentele au in partea stanga o supercheie (asa cum s-a mentionat orice cheie este in acelasi timp si supercheie)

F. Radulescu. Curs: Baze de date

7

EXEMPLUL 3

- ◆ Relatia $\text{Produce} = \text{IdP, NumeP, Qty, IdF, NumeF, AdresaF}$ avand dependentele:
- ◆ $F = \{ \text{IdP} \rightarrow \text{NumeP}, \text{IdP} \rightarrow \text{Qty}, \text{IdP} \rightarrow \text{IdF}, \text{IdF} \rightarrow \text{NumeF}, \text{IdF} \rightarrow \text{AdresaF} \}$
- ◆ Nu este in forma normala Boyce-Codd: cheia unica este IdP dar exista dependente care nu au in partea stanga o dupercheie: $\text{IdF} \rightarrow \text{NumeF}, \text{IdF} \rightarrow \text{AdresaF}$

F. Radulescu. Curs: Baze de date

8

FN3 – ATRIBUT PRIM

- ◆ Pentru definitia formei normale 3 este necesara definirea notiunii de **atribut prim**:
- ◆ **Definitie.** R o schema de relatie si F multimea de dependente functionale asociata. Un atribut $A \in R$ se numeste atribut prim daca el apartine unei chei a lui R.
- ◆ Exemplu: $R = \text{ABCDE}$ avand
- ◆ $F = \{ A \rightarrow B, B \rightarrow A, A \rightarrow C, D \rightarrow E \}$.
- ◆ Cum cheile relatiei sunt AD si BD rezulta ca in R sunt trei attribute prime: A, B si D.

F. Radulescu. Curs: Baze de date

9

FN3 - DEFINITIE

- ◆ **Definitie.** R o schema de relatie si F multimea de dependente functionale asociata. Se spune ca R este in forma normala 3 (FN3) daca si numai daca oricare ar fi o dependenta netriviala $X \rightarrow A$ din F atunci
- ◆ X este supercheie pentru R
- sau
- ◆ A este atribut prim

F. Radulescu. Curs: Baze de date

10

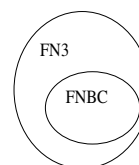
FN3 – cont.

- ◆ Daca in F avem dependente care contin mai multe attribute in partea dreapta putem aplica regula de descompunere pentru a obtine dependente care in partea dreapta au cate un singur atribut.
- ◆ Conditia de FNBC este inclusa in definitia FN3. Din acest motiv orice relatie care este in FNBC este implicit si in FN3. Reciproca nu este adevarata.
- ◆ De asemenea daca o schema de relatie nu este in FN3 ea nu poate fi nici in FNBC.

F. Radulescu. Curs: Baze de date

11

FN3 INCLUDE FNBC



F. Radulescu. Curs: Baze de date

12

EXEMPLUL 1

- ◆ Relatia $R = ABCD$ avand
- ◆ $F = \{ AB \rightarrow C, AB \rightarrow D, D \rightarrow A \}$ are cheia unica AB .
- ◆ Relatia este in FN3 deoarece primele doua dependente au in partea stanga o supercheie (AB) iar a treia dependenta are in partea dreapta atributul prim A .
- ◆ Relatia nu este in FNBC deoarece a treia dependenta violeaza definitia pentru aceasta forma normala (nu are in partea stanga o supercheie).

F. Radulescu. Curs: Baze de date

13

EXEMPLUL 2

- ◆ Relatia $R = ABCDE$ avand
- ◆ $F = \{ A \rightarrow B, B \rightarrow A, A \rightarrow C, D \rightarrow E \}$ are cheile AD si BD .
- ◆ R nu este in FN3 deoarece dependentele 3 si 4 nu au nici supercheie in partea stanga nici atribut prim in partea dreapta
- ◆ R nu e in FNBC deoarece nu e in FN3

F. Radulescu. Curs: Baze de date

14

EXEMPLUL 3

- ◆ Relatia $R = ABCD$ avand
- ◆ $F = \{ A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow A \}$ are cheile A, B, C si D . Rezulta ca:
- ◆ R este in FNBC deoarece in partea stanga a dependentelor sunt numai superchei
- ◆ R este in FN3 deoarece este in FNBC

F. Radulescu. Curs: Baze de date

15

FN1 si FN2

- ◆ Aceste forme normale nu garanteaza eliminarea anomaliilor deci ele nu sunt de dorit pentru schemele de relatie ale unei baze de date a unei aplicatii. Prezentam pe scurt definitia lor.

F. Radulescu. Curs: Baze de date

16

FN1

- ◆ **Definitie:** O relatie R este in **forma normala 1** (FN1) daca pe toate attributele sale exista doar valori atomice ale datelor.
- ◆ Semnificatia termenului 'atomic' este similara cu cea de la modelul entitate asociere: valoarea respectiva este intotdeauna folosita ca un intreg si nu se utilizeaza niciodata doar portiuni din aceasta.

F. Radulescu. Curs: Baze de date

17

FN1 – cont.

- ◆ De exemplu, daca intr-o relatie continand date despre persoane avem atributul **Adresa** acesta este atomic daca niciodata nu este nevoie sa fie folosite doar anumite portiuni ale sale (strada, numar, etc).

F. Radulescu. Curs: Baze de date

18

DEP. PARTIALE SI TRANZITIVE

- ◆ Fiind data o relatie R si multimea de dependente functionale asociata F putem defini inca doua concepte. Fie A un atribut neprim si X o multime de attribute din R. Atunci:
- ◆ **Definitie:** O dependenta functionala $X \rightarrow A$ se numeste **dependenta partiala** daca X este strict inclusa intr-o cheie a relatiei R.
- ◆ **Definitie:** O dependenta functionala $X \rightarrow A$ se numeste **dependenta tranzitiva** daca X nu este inclusa in nici o cheie a relatiei R.

F. Radulescu. Curs: Baze de date

19

FN2 - DEFINITIE

Definitie:

- ◆ Fie R o schema de relatie si F multimea de dependente functionale asociata.
- ◆ Se spune ca R este in forma normala 2 (FN2) daca si numai daca nu are dependente partiale (dar poate contine dependente tranzitive).
- ◆ Detalieri: O astfel de dependenta partiala nu exista in F si nici nu se poate deduce din F.
- ◆ Observatie: Daca R are chei formate dintr-un singur atribut, R e in FN2!

F. Radulescu. Curs: Baze de date

20

EXEMPLU

- ◆ $\text{Produce} = \text{IdP}, \text{NumeP}, \text{Qty}, \text{IdF}, \text{NumeF}, \text{AdresaF}$
- ◆ Multimea de dependente asociata este:
 $F = \{ \text{IdP} \rightarrow \text{NumeP}, \text{IdP} \rightarrow \text{Qty}, \text{IdP} \rightarrow \text{IdF}, \text{IdF} \rightarrow \text{NumeF}, \text{IdF} \rightarrow \text{AdresaF} \}$
- ◆ Este in FN2 pentru ca ultimele doua dependente sunt tranzitive dar nu sunt partiale (IdF nu apartine cheii unice IdP).

F. Radulescu. Curs: Baze de date

21

EXEMPLU - cont

Fie relatia:

Sala	Activitate	NrLocuri
EC 105	Curs SO	120
EC 104	Curs BD	90
EC 104	Curs POO	90
EC 105	Curs EA	120

Cheia relatiei: {Sala, Activitate}. Relatia nu este in FN2 deoarece exista dependenta $\text{Sala} \rightarrow \text{NrLocuri}$ care este partiala (Sala apartine unei chei dar nu e cheie)

F. Radulescu. Curs: Baze de date

22

ATENTIE!

- ◆ Asa cum s-a specificat anterior FN2 nu este o forma normala 'buna', ea trebuie evitata.
- ◆ Relatia din exemplul anterior prezinta toate anomaliiile enumerate la inceputul acestui capitol.

F. Radulescu. Curs: Baze de date

23

DESCOMPUNEREA SCHEMELOR DE RELATIE

- ◆ Asa cum s-a mentionat anterior, in cazul in care o relatie din baza de date nu este intr-o forma normala buna (FNBC, FN3) pot sa apara diverse anomalii.
- ◆ Solutia este inlocuirea relatiei respective cu doua sau mai multe relatii care sa contina aceleasi informatii dar care, fiecare in parte, este in forma normala dorita de proiectant.

F. Radulescu. Curs: Baze de date

24

DEFINITIE

- ◆ Procesul prin care se 'sparge' o relatie in mai multe relatii se numeste **descompunerea unei scheme de relatie**.

Formal putem defini acest concept astfel:

- ◆ **Definitie:** Fie R o schema de relatie, $R = A_1 A_2 \dots A_m$.

Se spune ca $\rho = (R_1, R_2, \dots, R_n)$ este o **descompunere** a lui R daca si numai daca

$$R = R_1 \cup R_2 \cup \dots \cup R_n$$

F. Radulescu. Curs: Baze de date

25

OBSERVATII

- ◆ Schemele R_1, R_2, \dots, R_n contin deci atribute din R, fiecare atribut A_i al schemei initiale trebuind sa se regaseasca in cel putin una dintre ele.
- ◆ Nu este necesar ca schemele sa fie disjuncte (in practica ele au aproape intotdeauna atribute comune).
- ◆ In exemplele de mai jos sunt prezentate cateva descompuneri valide ale unor scheme de relatii (unele insa incorecte din punct de vedere al pastrarii datelor si/sau dependentelor initiale)

F. Radulescu. Curs: Baze de date

26

EXEMPLUL 1

- ◆ Fie relatia R = ABCDE avand $F = \{ A \rightarrow B, B \rightarrow A, A \rightarrow C, D \rightarrow E \}$.
- ◆ Putem avea descompuneri ca:
 - ◆ $\rho_1 = (ABC, DE),$
 - ◆ $\rho_2 = (ABCD, DE),$
 - ◆ $\rho_3 = (AB, CD, DE)$

F. Radulescu. Curs: Baze de date

27

EXEMPLUL 2

- ◆ Fie relatia Produse = IdP, NumeP, Qty, IdF, NumeF, AdresaF avand dependentele functionale:
- ◆ $F = \{ IdP \rightarrow NumeP, IdP \rightarrow Qty, IdP \rightarrow IdF, IdF \rightarrow NumeF, IdF \rightarrow AdresaF \}$
- Putem avea o multitudine de descompuneri printre care:
 - ◆ $\rho_1 = ((IdP, NumeP, Qty, IdF); (NumeF, AdresaF))$
 - ◆ $\rho_2 = ((IdP, NumeP, Qty, IdF); (IdF, NumeF, AdresaF))$
 - ◆ $\rho_3 = ((IdP, NumeP); (Qty, IdF); (NumeF, AdresaF))$

F. Radulescu. Curs: Baze de date

28

SCHEMA si CONTINUT

- ◆ Descompunerea actioneaza deci la nivelul **schemei** relatiei. Ce se intampla insa cu **continutul** acesteia in cazul unei descompuneri?
- ◆ Fiecare relatie rezultata va mosteni o parte dintre datele relatiei descompuse si anume proiectia acesteia pe multimea de atribute a relatiei rezultata din descompunere.

F. Radulescu. Curs: Baze de date

29

Reamintire: PROIECTIA

A	B	C	D	E
1	1	2	1	3
2	1	2	1	3
2	7	4	4	1
2	3	9	2	1
1	3	7	4	1
1	3	9	2	1

Relatia R

B	C	E
1	2	3
7	4	1
3	9	1
3	7	1

Rezultatul proiectiei $\pi_{B,C,E}(R)$
Observam ca s-au eliminat doua linii duplicate din rezultat (cele provenite din liniile 2 si 6).

F. Radulescu. Curs: Baze de date

30

SCHEMA si CONTINUT

- ◆ Sa consideram o instanta r a relatiei de schema R (instanta unei relatii este o incarcare cu date corecte a acesteia).
- ◆ Atunci instantele pentru relatiile din descompunerea ρ sunt:

$$r_i = \pi_{R_i}(r)$$

F. Radulescu. Curs: Baze de date

31

EXEMPLU

IdP	NumeP	Qty	IdF	NumeF	AdresaF
101	Imprimanta laser	30	20	Xerox	Str. Daniel Danielopolu 4-6, Sector 1, Bucuresti
105	Calculator PC	20	23	IBM	Bd. D.Cantemir nr.1, Bucuresti
124	Copiator	10	20	Xerox	Str. Daniel Danielopolu 4-6, Sector 1, Bucuresti

F. Radulescu. Curs: Baze de date

32

$$\rho_1 = ((IdP, NumP, Qty, IdF); (NumeF, AdresaF))$$

IdP	NumeP	Qty	IdF
101	Imprimanta laser	30	20
105	Calculator PC	20	23
124	Copiator	10	20

NumeF	AdresaF
Xerox	Str. Daniel Danielopolu 4-6, Sector 1, Bucuresti
IBM	Bd. D.Cantemir nr.1, Bucuresti

F. Radulescu. Curs: Baze de date

33

$$\rho_2 = ((IdP, NumP, Qty, IdF); (IdF, NumeF, AdresaF))$$

IdP	NumeP	Qty	IdF
101	Imprimanta laser	30	20
105	Calculator PC	20	23
124	Copiator	10	20

IdF	NumeF	AdresaF
20	Xerox	Str. Daniel Danielopolu 4-6, Sector 1, Bucuresti
23	IBM	Bd. D.Cantemir nr.1, Bucuresti

F. Radulescu. Curs: Baze de date

34

$$\rho_3 = ((IdP, NumP); (Qty, IdF); (NumeF, AdresaF))$$

IdP	NumeP
101	Imprimanta laser
105	Calculator PC
124	Copiator

Qty	IdF
30	20
20	23
10	20

NumeF	AdresaF
Xerox	Str. Daniel Danielopolu 4-6, Sector 1, Bucuresti
IBM	Bd. D.Cantemir nr.1, Bucuresti

F. Radulescu. Curs: Baze de date

35

REZULTAT

- ◆ Observam din aceste exemple ca in cazul primei si ultimei descompuneri nu putem reconstrui prin join sau alti operatori relationali relatia initiala.
- ◆ In cazul in care descompunerea nu s-a facut corect putem pierde:
 - ◆ Datele relatiei initiale
 - ◆ Dependentele functionale ale relatiei initiale.
- ◆ In paragrafele urmatoare sunt prezentati algoritmi prin care putem detecta daca prin descompunere se pierd date sau dependente.

F. Radulescu. Curs: Baze de date

36

JOIN FARA PIERDERI

- ◆ Conditia pentru a nu se pierde date prin descompunere este ca relatia initiala sa poata fi reconstruita exact prin joinul natural al relatiilor rezultate, fara tupluri in minus sau in plus.
- ◆ Formal, definitia este urmatoarea:

F. Radulescu. Curs: Baze de date

37

JOIN FARA PIERDERI (2)

Definitie:

- ◆ Fie R o schema de relatie,
 - ◆ F multimea de dependente functionale asociata si
 - ◆ o descompunere $\rho = (R_1, R_2, \dots, R_n)$ a lui R.
- Se spune ca ρ este o descompunere **cu join fara pierderi in raport cu F** (prescurtat j.f.p.) daca si numai daca pentru orice instanta r a lui R care satisface dependentele F avem ca:

$$r_1 \bowtie r_2 \bowtie \dots \bowtie r_n = r \quad \text{unde} \\ r_i = \pi_{R_i}(r)$$

F. Radulescu. Curs: Baze de date

38

JOIN FARA PIERDERI (3)

- ◆ In exemplul anterior doar descompunerea $\rho_2 = (IdP, NumeP, Qty, IdF); (IdF, NumeF, AdresaF)$ are aceasta proprietate, in cazul celorlalte, din cauza inexistentiei coloanelor comune, joinul natural nu se poate efectua.

F. Radulescu. Curs: Baze de date

39

JOIN FARA PIERDERI (4)

- ◆ Faptul ca o descompunere are sau nu aceasta proprietate se poate testa pornind doar de la
 - ◆ lista atributelor relatiei initiale,
 - ◆ lista atributelor relatiilor din descompunere si
 - ◆ multimea de dependente functionale asociata
- ◆ Este deci o proprietate a schemei relatiei si nu a instantelor sale

F. Radulescu. Curs: Baze de date

40

ALGORITM TESTARE JFP

- ◆ **Intrare:** Schema de relatie $R = A_1 A_2 \dots A_m$, multimea de dependente functionale F si o descompunere $\rho = (R_1, R_2, \dots, R_n)$
- ◆ **Iesire:** Verdictul daca ρ are sau nu proprietatea j.f.p.
- ◆ **Metoda:**
Se construiesc o tabela avand n linii si m coloane. Liniile sunt etichetate cu elementele lui ρ iar coloanele cu attributele lui R.
Elementul (i,j) al tablei va fi egal cu
 - ◆ a_j daca $A_j \in R_i$ sau
 - ◆ b_{ij} altfel.

F. Radulescu. Curs: Baze de date

41

ALGORITM – cont.

- ◆ Se parcurg dependentele $X \rightarrow Y$ din F. Daca doua (sau mai multe) linii din tabela au aceiasi simbolii pe coloanele X aceste linii se egaleaza si pe coloanele din Y astfel:
 - Daca pe o coloana din Y apare un a_j atunci toate elementele de pe acea coloana din liniile respective devin a_j
 - Daca pe o coloana din Y nu apare nici un a_j atunci se alege unul dintre elementele de tip b_{ij} si toate elementele de pe acea coloana din liniile respective devin egale cu acel b_{ij}

F. Radulescu. Curs: Baze de date

42

ALGORITM – cont.

Procesul se opreste:

- ◆ Fie cand s-a obtinut o linie in tabela care contine doar a-uri, caz in care descompunerea ρ **are** proprietatea j.f.p.
- ◆ Fie cand la o parcurgere a dependentelor nu mai apar schimbari in tabela si nu s-a obtinut o linie doar cu a-uri. In acest caz descompunerea ρ **nu are** proprietatea j.f.p.

F. Radulescu. Curs: Baze de date

43

ALGORITM – cont.

- ◆ In literatura de specialitate se poate gasi demonstratia faptului ca acest algoritm determina corect daca o descompunere are proprietatea j.f.p. sau nu.

F. Radulescu. Curs: Baze de date

44

EXEMPLUL 1

- ◆ Exemplul 1: Fie $R = ABCDE$, $F = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D, D \rightarrow E \}$ si o descompunere a lui R $\rho = (ABCD, DE)$
- ◆ Construim tabelul din algoritm:

	A	B	C	D	E
ABCD	a1	a2	a3	a4	b1 5 a5
DE	b21	b22	b23	a4	a5

F. Radulescu. Curs: Baze de date

45

EXEMPLUL 1 – cont.

- ◆ La prima parcurgere, pentru dependentele $A \rightarrow B$, $A \rightarrow C$, $A \rightarrow D$ nu gasim doua linii cu aceleasi valori pe coloana A
- ◆ Pentru dependenta $D \rightarrow E$ cele doua linii sunt egale pe coloana D (simbolul ~~a4~~).
- ◆ Le egalam si pe coloana E: cum pe aceasta coloana exista ~~a5~~ rezulta ca ~~b1~~5 devine egal cu ~~a5~~.
- ◆ S-a obtinut o linie numai cu a-uri, deci descompunerea are proprietatea de join fara pierderi.

	A	B	C	D	E
ABCD	a1	a2	a3	a4	b1 5 a5
DE	b21	b22	b23	a4	a5

F. Radulescu. Curs: Baze de date

46

EXEMPLUL 2

- ◆ Fie relatia $R = ABCDE$, $\rho = (AB, BC, CDE)$.
- ◆ $F = \{ A \rightarrow B, AC \rightarrow D, D \rightarrow E \}$
- ◆ La prima trecere nu apar modificari in tabel. Procesul se opreste si ρ nu are proprietatea j.f.p.

	A	B	C	D	E
AB	a1	a2	b13	b14	b15
BC	b21	a2	a3	b24	b25
CDE	b31	b32	a3	a4	a5

F. Radulescu. Curs: Baze de date

47

EXEMPLUL 3

- ◆ Fie $R = ABCDE$, $\rho = (BCE, AB, ACD)$
- ◆ $F = \{ C \rightarrow E (1), A \rightarrow C (2), B \rightarrow D (3), D \rightarrow E (4), E \rightarrow B (5) \}$ (dep. numerotate intre paranteze)

	A	B	C	D	E
BCE	b11	a2	a3	b14	a5
AB	a1	a2	b23 a3 (2)	b24 b14 (3)	b25 a5 (4)
ACD	a1	b32 a2 (5)	a3	a4	b35 a5 (1)

F. Radulescu. Curs: Baze de date

48

EXEMPLUL 3 – cont.

- ◆ Prima trecere:
- ◆ Din $C \rightarrow E$ b_{35} devine a_5
- ◆ Din $A \rightarrow C$ b_{23} devine a_3

	A	B	C	D	E
BCE	b11	a2	a3	b14	a5
AB	a1	a2	b23 a3 (2)	b24 b14 (3)	b25 a5 (4)
ACD	a1	b32 a2 (5)	a3	a4	b35 a5 (1)

F. Radulescu. Curs: Baze de date

49

EXEMPLUL 3 – cont.

- ◆ Din $B \rightarrow D$ b_{24} devine b_{14}
- ◆ Din $D \rightarrow E$ b_{25} devine a_5
- ◆ Din $E \rightarrow B$ b_{32} devine a_2
- ◆ linie doar cu a-uri => j.f.p.

	A	B	C	D	E
BCE	b11	a2	a3	b14	a5
AB	a1	a2	b23 a3 (2)	b24 b14 (3)	b25 a5 (4)
ACD	a1	b32 a2 (5)	a3	a4	b35 a5 (1)

F. Radulescu. Curs: Baze de date

50

OBSERVATII

- ◆ In exemplele de mai sus a fost suficienta o singura trecere prin dependente.
- ◆ Exista insa situatii cand sunt necesare mai multe treceri pana procesul se opreste.
- ◆ In cazul in care descompunerea are doar doua elemente se poate testa daca are proprietatea de join fara pierderi si altfel

F. Radulescu. Curs: Baze de date

51

TEST JFP

- ◆ Fie R o schema de relatie,
 - ◆ F multimea de dependente functionale asociata si
 - ◆ $\rho = (R_1, R_2)$ o descompunere a sa.
- Atunci ρ are proprietatea de join fara pierderi daca una din dependentele urmatoare se poate deduce din F :
- ◆ $(R_1 \cap R_2) \rightarrow (R_1 - R_2)$ sau
 - ◆ $(R_1 \cap R_2) \rightarrow (R_2 - R_1)$

F. Radulescu. Curs: Baze de date

52

TEST JFP – cont.

- ◆ Pentru a testa daca dependenta se poate deduce din F este suficient sa calculam inchiderea lui $(R_1 \cap R_2)$.
- ◆ Daca ea contine fie pe $(R_1 - R_2)$ fie pe $(R_2 - R_1)$ atunci descompunerea este cu join fara pierderi.

F. Radulescu. Curs: Baze de date

53

EXEMPLU

- ◆ Fie $R = ABCDE$,
- ◆ $F = \{ A \rightarrow B, A \rightarrow C, A \rightarrow D, D \rightarrow E \}$
- ◆ $\rho = (ABCD, DE)$.

Avem $R_1 = ABCD$, $R_2 = DE$. Rezulta ca:

- ◆ $(R_1 - R_2) = ABCD - DE = ABC$
- ◆ $(R_2 - R_1) = DE - ABCD = E$
- ◆ $(R_1 \cap R_2) = D$

F. Radulescu. Curs: Baze de date

54

EXEMPLU – cont.

Cele doua dependente sunt:

$$\blacklozenge (R_1 \cap R_2) \rightarrow (R_1 - R_2) \text{ devine} \\ D \rightarrow ABC$$

$$\blacklozenge (R_1 \cap R_2) \rightarrow (R_2 - R_1) \text{ devine} \\ D \rightarrow E$$

Ultima este chiar o dependenta din F deci se poate deduce din F deci ρ are proprietatea de join fara pierderi.

F. Radulescu. Curs: Baze de date

55

PASTRARE DEPENDENTE

◆ O a doua problema in cazul descompunerii unei scheme R avand dependentele F in mai multe relatii R_1, R_2, \dots, R_n este aceea a pastrarii corelatiilor intre date, corelatii date de dependente functionale din F.

◆ Fiecare relatie R_i va mosteni o multime de dependente data de proiectia multimii de dependente functionale F pe R_i

$$F_i = \pi_{R_i}(F)$$

F. Radulescu. Curs: Baze de date

56

EXEMPLU

◆ Fie relatia Produse = IdP, NumeP, Qty, IdF, NumeF, AdresaF avand dependentele functionale:

$$\blacklozenge F = \{ \text{IdP} \rightarrow \text{NumeP}, \text{IdP} \rightarrow \text{Qty}, \text{IdP} \rightarrow \text{IdF}, \\ \text{IdF} \rightarrow \text{NumeF}, \text{IdF} \rightarrow \text{AdresaF} \}$$

◆ In cazul descompunerii $\rho_2 = (R_1, R_2)$ unde:

$$\blacklozenge R_1 = (\text{IdP}, \text{NumeP}, \text{Qty}, \text{IdF})$$

$$\blacklozenge R_2 = (\text{IdF}, \text{NumeF}, \text{AdresaF})$$

cele doua relatii mostenesc urmatoarele dependente:

$$F_{R_1} = \pi_{R_1}(F) = \{ \text{IdP} \rightarrow \text{NumeP}, \text{IdP} \rightarrow \text{Qty}, \text{IdP} \rightarrow \text{IdF} \}$$

$$F_{R_2} = \pi_{R_2}(F) = \{ \text{IdF} \rightarrow \text{NumeF}, \text{IdF} \rightarrow \text{AdresaF} \}$$

F. Radulescu. Curs: Baze de date

57

PASTRARE DEPENDENTE (2)

◆ Dupa cum se observa toate dependentele relatiei initiale sunt pastrate fie in F_{R_1} fie in F_{R_2} .

◆ Exista insa si cazuri in care unele dependente din F nu mai pot fi regasite in multimile de dependente asociate schemelor din descompunere si nu se pot deduce din acestea.

◆ In primul caz se spune ca descompunerea pastreaza dependentele iar in al doilea ca descompunerea nu pastreaza dependentele.

F. Radulescu. Curs: Baze de date

58

DEFINITIE

◆ Fie R o schema de relatie, F multimea de dependente functionale asociata, o descompunere $\rho = (R_1, R_2, \dots, R_n)$ a lui R si

$$F_i = \pi_{R_i}(F) \text{ multimile de dependente functionale ale elementelor descompunerii.}$$

◆ Se spune ca ρ **pastreaza dependentele** din F daca si numai daca orice dependenta din F poate fi dedusa din:

$$\cup_{i=1..n} (F_i).$$

F. Radulescu. Curs: Baze de date

59

DEFINITIE – cont.

◆ Rezulta ca o descompunere pastreaza dependentele daca si numai daca:

$$F \subseteq (\cup_{i=1..n} (F_i))^+$$

◆ Din pacate atat proiectia unei multimi de dependente cat si incluziunea de mai sus implica un calcul de inchidere a unei multimi de dependente (F si respectiv reuniunea multimilor F_i).

◆ Exista si in acest caz un algoritm pentru a testa daca o dependenta este sau nu pastrata dupa descompunere fara a fi necesar efectiv calculul multimilor F_i

F. Radulescu. Curs: Baze de date

60

ALGORITM DE TESTARE A PASTRARII DEPENDENTELOR

- ◆ **Intrare:** o schema de relatie R, multimea de dependente functionale asociata F si o descompunere $\rho = (R_1, R_2, \dots, R_n)$
- ◆ **Iesire:** verdictul daca ρ pastreaza sau nu dependentele
- ◆ **Metoda:** Pentru fiecare dependenta $X \rightarrow Y$ din F se procedeaza astfel:

F. Radulescu. Curs: Baze de date

61

ALGORITM – cont.

- ◆ Se porneste cu o multime de atribute Z
 $Z = X$
- ◆ Se parcurg repetat elementele descompunerii ρ .
- ◆ Pentru fiecare R_i se calculeaza o noua valoare a lui Z astfel:
- ◆ $Z = Z \cup ((Z \cap R_i)^+ \cap R_i)$

F. Radulescu. Curs: Baze de date

62

ALGORITM – cont.

- ◆ Procesul se opreste in momentul in care Z ramane neschimbat la o parcurgere a elementelor R_i .
- ◆ Daca $Y \subseteq Z$ atunci dependenta $X \rightarrow Y$ este pastrata, altfel nu e pastrata

Daca toate dependentele din F sunt pastrate inseamna ca ρ pastreaza dependentele din F.

F. Radulescu. Curs: Baze de date

63

EXEMPLUL 1

- ◆ Fie $R = ABCDE$,
 - ◆ $\rho = (BCE, AB, ACD)$
 - ◆ $F = \{ C \rightarrow E, A \rightarrow C, B \rightarrow D, D \rightarrow E, E \rightarrow B \}$
- Se observa ca dependentele $C \rightarrow E, A \rightarrow C$ si $E \rightarrow B$ sunt pastrate: ele apartin proiectiei lui F pe BCE (prima si ultima) si ACD (a doua).
- ◆ Raman de testat dependentele $B \rightarrow D$ si $D \rightarrow E$. Sa aplicam algoritmul pentru $B \rightarrow D$:

F. Radulescu. Curs: Baze de date

64

EXEMPLUL 1 – cont.

- ◆ Initial $Z = B$
Trecerea 1 prin elementele lui ρ :
- ◆ Pentru BCE:
 $Z = B \cup ((B \cap BCE)^+ \cap BCE) =$
 $B \cup (BDE \cap BCE) = BE$
- ◆ Pentru AB:
 $Z = BE \cup ((BE \cap AB)^+ \cap AB) =$
 $BE \cup (BDE \cap AB) = BE$
- ◆ Pentru ACD:
 $Z = BE \cup ((BE \cap ACD)^+ \cap AB) = BE \cup \emptyset = BE$

F. Radulescu. Curs: Baze de date

65

EXEMPLUL 1 – cont.

- ◆ La urmatoarea trecere Z ramane neschimbat si procesul se opreste.
- ◆ Cum $\{D\} \not\subseteq BE$ rezulta ca dependenta $B \rightarrow D$ nu este pastrata deci ρ nu pastreaza dependentele

F. Radulescu. Curs: Baze de date

66

EXEMPLUL 2

- ◆ Fie schema de relatie $R = ABCD$,
- ◆ $F = \{ A \rightarrow B, A \rightarrow C, C \rightarrow D, D \rightarrow A \}$
- ◆ $\rho = (ABC, CD)$.

Trebuie sa testam daca $D \rightarrow A$ este pastrata (celelalte dependente se regasesc direct in proiectiile lui F pe elementele descompunerii).

F. Radulescu. Curs: Baze de date

67

EXEMPLUL 2 – cont.

- ◆ Initial $Z = D$

Prima trecere prin elementele lui ρ :

- ◆ Pentru ABC :

$$Z = D \cup ((D \cap ABC)^+ \cap ABC) = \\ D \cup \emptyset = D$$

- ◆ Pentru CD :

$$Z = D \cup ((D \cap CD)^+ \cap CD) = \\ D \cup (ABCD \cap CD) = CD$$

F. Radulescu. Curs: Baze de date

68

EXEMPLUL 2 – cont.

- ◆ A doua trecere prin elementele lui ρ :

- ◆ Pentru ABC :

$$Z = CD \cup ((CD \cap ABC)^+ \cap ABC) = \\ CD \cup (ABCD \cap ABC) = ABCD.$$

- ◆ Stop. Am obtinut ca $A \subseteq Z$, deci dependenta $D \rightarrow A$ este pastrata, deci ρ pastreaza dependentele.

F. Radulescu. Curs: Baze de date

69

Sfarsitul cursului 8:
Proiectarea bazelor de date
relationale - partea 2

F. Radulescu. Curs: Baze de date

70