UNIUNEA EUROPEANĂ

GUVERNUL ROMÂNIEI

Instrumente Structurale
2007-2013

Platformă de e-learning și curriculă e-content
pentru învățământul superior tehnic

Baze de date 1

11. Normalizarea bazelor de date

PROBLEMATICA

- O categorie de probleme care pot sa apara in dezvoltarea unei aplicatii continand o baza de date este cea a proiectarii incorecte a schemelor de relatie.
- In acest caz pot sa apara o serie de ***anomalii*** care pot complica procesul de programare.
- Testarea corectitudinii unei scheme de relatie poate fi facuta cu ajutorul ***dependentelor functionale - DF*** (sau de alt tip) atasate acelei scheme.

PROBLEMATICA – CONT.

- DF modeleaza corelatii care exista intre datele din lumea reala stocate in baza de date si reprezinta criterii de corectitudine ale datelor incarcate in baza de date.
- In cazul in care o relatie nu are o schema corespunzatoare ea trebuie inlocuita cu doua sau mai multe relatii (operatia este numita si ***descompunerea unei scheme de relatie***), fiecare relatie rezultata avand o schema corecta – aflata in ***forma normala*** dorita.

ANOMALII (Cap 2)

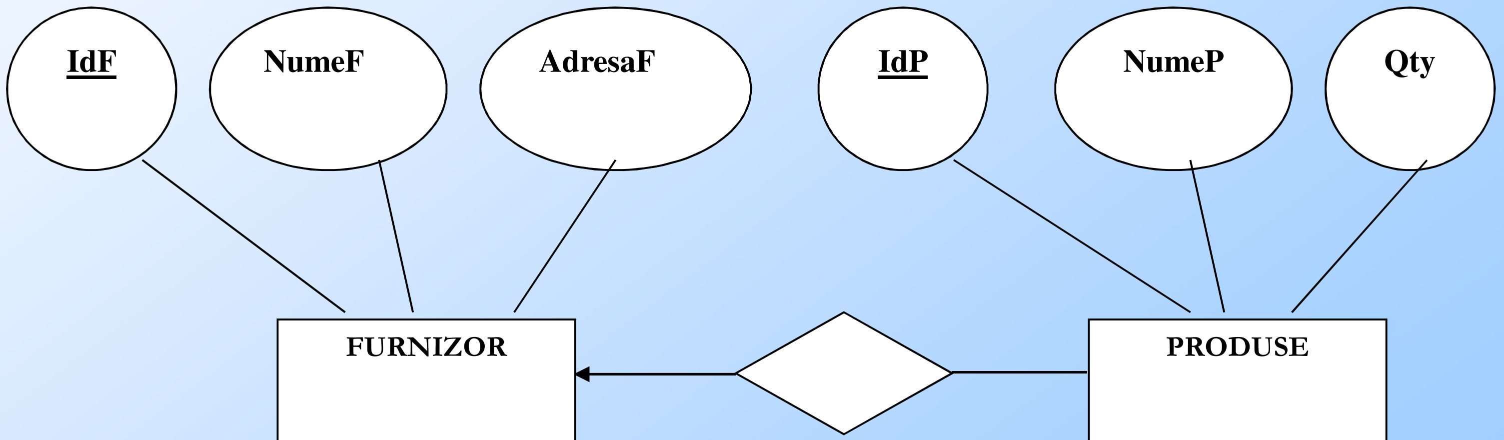
IDP	NUMEP	QTY	IDF	NUMEF	ADRESAF
101	Imprimantă laser	30	20	XY SRL	Str. X, Bucureşti
105	Calculator PC	20	23	Z SRL	Bd. Z, Bucureşti
124	Copiator	10	20	XY SRL	Str. X, Bucureşti

ANOMALII

- **Redundanta:** Redundanta reprezinta stocarea in mod nejustificata a unei aceleiasi informatii de mai multe ori in baza de date. Observam ca pentru fiecare produs este stocat numele si adresa furnizorului, desi ele sunt unic determinate de codul acestuia.
- **Anomalia de stergere:** La stergerea din relatie a ultimului produs al unui furnizor se pierd automat si datele despre acesta.
- **Anomalia de actualizare:** In cazul actualizarii unei informatii redundante, se poate intampla ca operatia sa modifice unele aparitii ale acesteia iar altele sa ramana cu vechea valoare.
- **Anomalia de inserare:** Nu putem insera date despre un furnizor (numele si adresa sa) decat daca exista in stoc un produs furnizat de acesta.

SURSA ANOMALIILOR DIN EXEMPLU

- Aceste anomalii apar in relatia PRODUSE deoarece intr-o aceeasi tabela au fost stocate date despre doua clase diferite de obiecte.
- In cazul proiectarii cu ajutorul modelului entitate-asociere diagrama corecta este urmatoarea:



REZULTAT TRANSFORMARE

Prin transformarea acestei diagrame se obtin urmatoarele scheme de relatie:

- Furnizor(<u>IdF</u>, NumeF, AdresaF)
- Produse(<u>IdP</u>, NumeP, Qty, IdF)

Tabelele FIRMA si PRODUS

IdF	NumeF	AdresaF
20	XY SRL	Str. X Bucureşti
23	Z SRL	Bd. Z, Bucuresti

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

OBIECTIVE DESCOMPUNERE

- Procesul de ‘spargere’ a unei tabele care are o structura incorecta in doua sau mai multe tabele se numeste ***descompunerea schemei de relatie***.
- Pentru detectarea relatiilor care trebuiesc descompuse exista o serie de reguli de corectitudine, numite si ***forme normale***.
- Definirea acestor forme normale se bazeaza pe notiunea de ***dependenta (functionala sau multivalorica)*** prezentata in continuare.

DEPENDENŢE FUNCŢIONALE

Definitie: Fie:

- R o schema de relatie
- X, Y $\subseteq$R doua multimi de atribute ale acesteia.

Spunem ca ***X determina functional pe Y*** (sau Y este determinata functional de X) daca si numai daca oricare ar fi doua tupluri t1 si t2 din <u>orice</u> instanta a lui R atunci:

$$t1[X] = t2[X] \Rightarrow t1[Y] = t2[Y].$$

DEPENDENŢE FUNCŢIONALE (2)

- Altfel spus, daca doua tupluri au aceleasi valori pe atributele X atunci ele au aceleasi valori si pe atributele Y.
- Notatia pentru dependente functionale este o sageata de la stanga spre dreapta:

$$X \rightarrow Y$$

EXEMPLU

- **Exemplu:** In relatia Produse din paragraful anterior putem scrie urmatoarele dependente functionale:
- IdP $\rightarrow$ NumeP, Qty, IdF, NumeF, AdresaF,
- IdF $\rightarrow$ NumeF, AdresaF

Aceste dependente arata ca

- daca doua produse au acelasi IdP, este vorba de fapt de acelasi produs
- daca doua produse au acelasi IdF (Id furnizor) atunci si valorile pentru numele si adresa acestuia trebuie sa fie aceleasi.

OBSERVATIE IMPORTANTA

- Dependentele functionale nu se determina din inspectarea continutului de la un moment dat al relatiei ci din semnificatia atributelor acesteia.
- In exemplul prezentat, a doua DF arata ca daca la doua produse apare acelasi Id furnizor atunci numele si adresa furnizorului sunt de asemenea aceleasi (deoarece nu pot sa existe doi furnizori diferiti cu acelasi Id).

AXIOME SI REGULI

- Pornind de la o multime de dependente functionale atasate unei scheme de relatie se pot deduce alte dependente functionale valide.
- Exista o multitudine de reguli de inferenta. Pentru a se putea face o prezentare formala a acestora, trei dintre ele au fost alese ca axiome iar restul se pot deduce pornind de la ele.
- Cele trei axiome (numite in literatura si ***Axiomele lui Armstrong***) sunt urmatoarele:

A1 - REFLEXIVITATEA

- **A1. Reflexivitate:** Fie R o schema de relatie si $X \subseteq R$. Atunci:

  Daca $Y \subseteq X$ atunci $X \rightarrow Y$

- Toate dependentele functionale care rezulta din aceast axioma sunt numite si ***dependente triviale***. Ele nu spun nimic in plus fata de setul de dependente initial dar sunt dependente functionale valide.

A2 - AUGMENTARE

- **A2. Augmentare:** Fie R o schema de relatie si X, Y, Z $\subseteq$ R. Atunci:

  Daca X $\rightarrow$ Y atunci si XZ $\rightarrow$ YZ

- Aceasta axioma arata ca se poate reuni o aceeasi multime Z in stanga si in dreapta unei dependente functionale valide obtinand de asemenea o dependenta functionala valida.

A3 - TRANZITIVITATE

- **A3. Tranzitivitate:** Fie R o schema de relatie si X, Y, Z $\subseteq$ R.

  Daca X $\rightarrow$ Y si Y $\rightarrow$ Z atunci si X $\rightarrow$ Z

REGULI

- Pe baza acestor axiome se pot demonstra o serie de reguli de inferenta pentru dependente functionale dintre care cele mai importante sunt urmatoarele:

R1 - DESCOMPUNERE

- **R1. Descompunere:** Fie R o schema de relatie si X, Y, Z $\subseteq$ R.

  Daca X $\rightarrow$ Y si Z $\subseteq$ Y atunci si X $\rightarrow$ Z
- Regula descompunerii ne permite sa rescriem un set de dependente functionale astfel incat sa obtinem doar dependente care au in partea dreapta doar un singur atribut.

R1 - DESCOMPUNERE – cont.

- Sa presupunem ca avem o dependenta functionala de forma:

$$X \rightarrow A_1A_2A_3...A_n$$

- Atunci ea poate fi inlocuita cu urmatoarele **n** dependente functionale:

$$X \rightarrow A_1$$
$$X \rightarrow A_2$$
$$X \rightarrow A_3$$
$$\cdots$$
$$X \rightarrow A_n$$

R2 - REUNIUNE

- **R2. Reuniune:** Fie R o schema de relatie si X, Y, Z $\subseteq$ R.

  Daca X $\rightarrow$ Y si X $\rightarrow$ Z atunci si X $\rightarrow$ YZ

- Rezulta si faptul ca din cele **n** reguli obtinute prin descompunere se poate obtine dependenta initiala, deci inlocuirea acesteia nu duce la pierderea vreunei corelatii existente.

R3 - PSEUDOTRANZITIVITATEA

- **R3. Pseudotranzitivitate:** Fie R o schema de relatie si X, Y, Z, W $\subseteq$ R.

Daca X $\rightarrow$ Y si YZ $\rightarrow$ W atunci si XZ $\rightarrow$ W

DEMONSTRATII

- **Exercitiu:** Demonstrati cele trei reguli folosind axiomele lui Armstrong.

Exemplu de demonstratie R3:

- Augmentam prima dependenta cu Z. Obtinem $XZ \rightarrow YZ$.
- Din aceasta dependenta si din $YZ \rightarrow W$ obtinem prin tranzitivitate $XZ \rightarrow W$, qed.

INCHIDEREA UNEI MULTIMI DE DF

- Pornind de la un set de dependente functionale F si utilizand axiomele si regulile obtinem o multitudine de alte dependente, triviale sau nu.
- Multimea tuturor dependentelor functionale care se pot deduce din F se numeste ***inchiderea multimii de dependente F***, notata cu F^+.

DEFINITIE FORMALA

- Definitia formala a acestei inchideri este urmatoarea:

$$F^{+} = \{X \rightarrow Y \mid F \Rightarrow X \rightarrow Y\}$$

- Prin $\Rightarrow$ am notat faptul ca dependenta respectiva de poate deduce din F folosind axiomele si regulile.

OBSERVATIE

- Multimea F^+ contine foarte multe dependente, inclusiv dependente triviale ca:
  - $ABC \rightarrow A$,
  - $ABC \rightarrow B$,
  - $ABC \rightarrow C$,
  - $ABC \rightarrow AB$,
  - $ABC \rightarrow AC$,
  - $ABC \rightarrow BC$ sau
  - $ABC \rightarrow ABC$

NU SE CALCULEAZA!

- Inchiderea unei multimi de dependente functionale nu se calculeaza, algoritmii care au nevoie de ea ocolind intr-un fel sau altul calculul acesteia.
- Introducerea acestei notiuni s-a facut pentru:
  - in cazul descompunerii unei scheme de relatie, aflarea dependentelor mostenite de la relatia initiala
  - pentru a putea defini formal alte notiuni

ACOPERIREA

- **Acoperirea unei multimi de DF:** Fie R o schema de relatie si F, G doua multimi de dependente pentru R. Se spune ca F ***acopera*** pe G daca si numai daca $G \subseteq F^+$.

ECHIVALENTA

- **Echivalenta a doua multimi de dependente:**
- Fie R o schema de relatie si F, G doua multimi de dependente pentru R.
- Se spune ca F ***e echivalenta*** cu G daca si numai daca F acopera pe G si G acopera pe F (deci $G \subseteq F^+$ si $F \subseteq G^+$, deci $F^+ = G^+$)

FORMA CANONICA

- **Forma canonica a unei multimi de DF:**
- Din definitiile de mai sus rezulta ca o multime de dependente poate fi inlocuita cu alta echivalenta continand alte dependente.
- In cazul in care aceasta multime indeplineste conditiile urmatoare se spune ca este in ***forma canonica***:

FORMA CANONICA – cont.

- Orice dependenta are in partea dreapta un singur atribut. Acest lucru se poate obtine aplicand regula descompunerii prezentata anterior.
- Multimea de dependente este minimala, nici una dintre dependente neputand sa fie dedusa din celelalte (nu exista dependente redundante).
- Partea dreapta a oricarei dependente este minimala (nici un atribut nu poate fi inlaturat fara ca asta sa duca la schimbarea lui F^+)

EXEMPLUL 1

- Fie R = ABCDE o schema de relatie si F multimea de dependente functionale asociata, cu $F = \{ AB \to CDE, C \to DE \}$:
- Aplicam regula de descompunere. Obtinem:
- $F = \{ AB \to C, AB \to D, AB \to E, C \to D, C \to E \}$
- Noua F nu e minimala deoarece $AB \to D$ si $AB \to E$ se pot deduce prin tranzitivitate din $AB \to C$ impreuna cu $C \to D$, $C \to E$.
- Rezulta ca forma canonica a lui F este:

$$F = \{ AB \to C, C \to D, C \to E \}$$

EXEMPLUL 2

- Pentru relatia

Produse(IdP, NumeP, Qty, IdF, NumeF, AdresaF, IdF)

din paragraful 4.1. avand multimea de dependente functionale:

F = { IdP $\rightarrow$ NumeP, Qty, IdF, NumeF, AdresaF;

IdF $\rightarrow$ NumeF, AdresaF}

EXEMPLUL 2 – cont.

- Forma canonica a lui F este:

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

EXEMPLUL 2 – cont.

Si in acest caz au fost eliminate doua dependente redundante:

- IdP $\rightarrow$ NumeF
- IdP $\rightarrow$ AdresaF

CHEIE

- **Definitie:** Fie R o schema de relatie, F multimea de dependente functionale asociata si $X \subseteq R$. Atunci X este cheie pentru R daca si numai daca:
  - $F \Rightarrow X \rightarrow R$ (deci $X \rightarrow R$ se poate deduce din F)

  si

  - X este minimala: oricare ar fi $Y \subset X$, $Y \neq X$ atunci $\neg(F \Rightarrow Y \rightarrow R)$ (deci orice submultime stricta a lui X nu mai indeplineste conditia anterioara).

CHEIE– cont.

- Deci:
  - o cheie determina functional toate atributele relatiei si
  - este minimala: nici o submultime stricta a sa nu determina functional pe R.
- Se observa faptul ca aceasta definitie este echivalenta cu cea din capitolul 3: cunoscandu-se valorile pe atributele X sunt unic determinate valorile pentru toate atributele relatiei, deci este unic determinat tuplul din relatie.

SUPERCHEIE

- In cazul in care doar prima conditie este indeplinita multimea X se numeste **supercheie**.
- **Observatie:** Faptul ca o supercheie nu este constransa de minimalitate nu inseamna insa ca ea nu poate fi minimala.
- Rezulta ca orice cheie este in acelasi timp si supercheie, reciproca nefiind insa adevarata.

EXEMPLU

- Fie R = ABCDE si F = { $AB \to C$, $C \to D$, $C \to E$ }. Atunci AB este cheie pentru R:
  - Din $AB \to C$, $C \to D$ si $C \to E$ obtinem prin tranzitivitate $AB \to D$ si $AB \to E$
  - Din $AB \to C$, $AB \to D$ si $AB \to E$ obtinem prin reuniune $AB \to CDE$
  - Din $AB \to CDE$ obtinem (augmentare cu AB) $AB \to ABCDE$, deci $AB \to R$
- Rezulta ca AB este supercheie pentru R. In paragraful urmator vom vedea cum se poate demonstra si faptul ca AB este minimala, deci este nu numai supercheie ci chiar cheie pentru R.

PROIECTIA UNEI MULTIMI DE DEPENDENTE FUNCTIONALE

- Asa cum s-a mentionat anterior inchiderea unei multimi de dependente functionale F^+ a fost introdusa si pentru a putea defini setul de dependente functionale mostenite de o schema de relatie obtinuta prin descompunerea unei scheme incorecte.

PROIECTIA … DF (2)

- Sa luam cazul relatiei anterioare continand produsele dintr-un depozit:

Produse = IdP, NumeP, Qty, IdF, NumeF, AdresaF

- Multimea de dependente asociata este:

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

PROIECTIA ... DF (3)

- Prin descompunerea acestei relatii in doua obtinem relatiile:

Produse = IdP, NumeP, Qty, IdF

Furnizori= IdF, NumeF, AdresaF

- Atributele relatiei initiale se regasesc fie doar intr-una dintre schemele rezultate fie in amandoua.
- Problema: ce dependente mostenesc cele doua relatii de la relatia initiala?

PROIECTIA ... DF (3)

- Solutia este de a defini proiectia unei multimi de dependente pe o multime de atribute.
- **Definitie.** Fie o relatie R, o multime asociata de dependente functionale F si o submultime de atribute $S \subseteq R$. ***Proiectia multimii de dependente F pe S***, notata cu $\pi_S(F)$ este multimea dependentelor din F^+ care au si partea stanga si pe cea dreapta incluse in S.
- Formal putem scrie:

$$\pi_S(F) = \{X \rightarrow Y \in F^+ \mid X, Y \subseteq S \}$$

EXEMPLU

Pentru exemplul de mai sus proiectiile sunt urmatoarele:

- $F_{PRODUSE} = \pi_{PRODUSE}(F) =$

$\{ IdP \rightarrow NumeP, IdP \rightarrow Qty, IdP \rightarrow IdF \}$

- $F_{FURNIZORI} = \pi_{FURNIZORI}(F) =$

$\{ IdF \rightarrow NumeF, IdF \rightarrow AdresaF \}$

OBSERVATIE

- **Observatie:** Atunci cand descompunem o schema se poate intampla ca unele dintre dependentele schemei initiale sa se piarda.
- Exemplu: Fie R = ABCD si F = { $A \rightarrow B$, $B \rightarrow C$, $C \rightarrow D$, $D \rightarrow A$ }. In cazul in care descompunem R in R1 = AB si R2 = CD atunci:

$$F_{R1} = \pi_{R1}(F) = \{ A \rightarrow B, B \rightarrow A \}$$

$$F_{R2} = \pi_{R2}(F) = \{ C \rightarrow D, D \rightarrow C \}$$

- A doua dependenta din fiecare multime nu este in F dar este in F^+ (obtinuta prin tranzitivitate).
- Observam insa ca dependentele $B \rightarrow C$ si $D \rightarrow A$ nu mai pot fi obtinute nici din F_{R1} nici din F_{R2} nici din reuniunea lor.

INCHIDEREA UNEI MULTIMI DE ATRIBUTE

- Fie R o schema de relatie, F multimea de dependente asociata si $X \subseteq R$. Se poate defini ***inchiderea multimii de atribute X in raport cu F*** (notata X^+) astfel:
- $X^+ = \{ A \mid X \rightarrow A \in F^+ \}$
- X^+ contine deci toate atributele care apar in partea dreapta a unei dependente din F sau care se poate deduce din F folosind regulile si axiomele si care il are in partea stanga pe X.

ALGORITM DE CALCUL X^+

- Intrare: R o schema de relatie, F multimea de dependente asociata si $X \subseteq R$
- Iesire: X^+
- Metoda: se procedeaza iterativ astfel:
  - Se porneste cu $X^{(0)} = X$
  - Pentru $i \geq 1$, $X^{(i)} = X^{(i-1)} \cup$

    $\{ A \mid (\exists)\ Y \rightarrow A \in F \text{ cu } Y \subseteq X^{(i-1)} \}$
  - Oprirea se face atunci cand $X^{(i)} = X^{(i-1)}$
  - Rezultat: $X^+ = X^{(i)}$

EXEMPLU

- Fie R = ABCDE si F = { $A \rightarrow B$, $A \rightarrow C$, $D \rightarrow E$ }. Pentru a calcula A^+ si AD^+ procedam astfel:

Calcul A^+ :

- $X^{(0)} = \{A\}$
- Din $A \rightarrow B$ si $A \rightarrow C$ rezulta ca $X^{(1)} = X^{(0)} \cup \{ B, C \} = \{ A \} \cup \{ B, C \} = ABC$
- Singurele dependente care au partea dreapta in $X^{(1)}$ sunt tot primele doua deci

  $X^{(2)} = X^{(1)} \cup \{ B, C \} = \{ A, B, C \} \cup \{ B, C \} = ABC$
- Oprire deoarece $X^{(2)} = X^{(1)}$
- Rezulta ca $(A)^+ = ABC$

EXEMPLU – cont.

Calcul AD⁺ :

- $X^{(0)} = \{A, D\}$
- Din $A \rightarrow B$, $A \rightarrow C$ si $D \rightarrow E$ rezulta ca $X^{(1)} = X^{(0)} \cup \{ B, C, E \} =$
  $\{ A, D \} \cup \{ B, C , E\} = ABCDE$
- Cum $X^{(1)} = R$ rezulta si ca $X^{(2)} = X^{(1)}$, deci oprire (oricate iteratii am face nu mai pot sa apara noi atribute).
- $(AD)^{+} = X^{(2)} = ABCDE$

INCHIDEREA … ATR. –cont.

- Scopul introducerii acestei notiuni (X^+) este si cel de a putea ocoli in alti algoritmi si definitii calculul lui F^+ . Avem urmatorul rezultat teoretic:
- **Propozitie:** Fie R o schema de relatie, F multimea de dependente asociata si X, Y $\subseteq$ R. Atunci X $\rightarrow$ Y se poare deduce din F daca si numai daca Y $\in$ X^+
- Demonstratia acestei propozitii se gaseste in literatura de specialitate.

ALTA DEFINITIE A CHEII

- Pe baza propozitiei din paragraful anterior se poate da o alta definitie pentru cheia sau supercheia unei relatii, bazata nu pe dependente functionale ca in paragraful anterior ci pe inchiderea unei multimi de atribute.

CHEIE - REAMINTIRE

- **Definitie:** Fie R o schema de relatie si $X \subseteq R$. Atunci X este cheie pentru R daca si numai daca:
  - $F \Rightarrow X \rightarrow R$ (deci $X \rightarrow R$ se poate deduce din F)

  si

  - X este minimala: oricare ar fi $Y \subset X$, $Y \neq X$ atunci $\neg(F \Rightarrow Y \rightarrow R)$ (deci orice submultime stricta a lui X nu mai indeplineste conditia anterioara).

ALTA DEFINITIE A CHEII (2)

- **Definitie:** Fie R o schema de relatie, F multimea de dependente functionale asociata si X $\subseteq$ R. Atunci X este cheie pentru R daca si numai daca:
  - $X^+ = R$

    si
  - X este minimala: oricare ar fi $Y \subset X$, $Y \neq X$ atunci $Y^+ \neq R$ (deci orice submultime stricta a lui X nu mai indeplineste conditia anterioara).
- Daca numai prima conditie este indeplinita atunci X este supercheie pentru R

ECHIVALENTA DEFINITII

Echivalenta acestei definitii cu cea anterioara este evidenta:

- $X^+ = R$ inseamna cf. propozitiei ca

  $X \rightarrow R$

- minimalitatea este de asemenea definita echivalent: $\neg(F \Rightarrow Y \rightarrow R)$ este echivalenta cu $\neg(Y^+ = R)$ adica $Y^+ \neq R$

GASIREA CHEILOR

- Folosind aceasta definitie se poate defini o euristica de gasire a cheilor unei relatii.
- Se cauta multimi minimale X care indeplinesc conditia $X^+ = R$
- Prezentam o euristica de gasire a cheilor:

NOTA IMPORTANTA

- Observatie: Atributele care nu apar in partea dreapta a nici unei dependente trebuie sa existe in orice cheie, ele neputand sa apara in procesul de calcul al inchiderii unei multimi de atribute.

EURISTICA

- Intrare: R o schema de relatie si F multimea de dependente functionale asociata (F in forma canonica).
- Iesire: Cheia unica sau cheile alternative ale lui R
- Metoda:

1. Se porneste de la multimea de atribute $X \subseteq R$ care nu apar in partea dreapta a nici unei dependente

EURISTICA (2)

2. Se calculeaza X^+. Daca $X^+ = R$ atunci X este cheia unica minimala a relatiei R si calculul se opreste aici. Pasii urmatori se efectueaza doar daca $X^+ \neq R$
3. Se adauga la X cate un atribut din $R - X^+$ obtinandu-se o multime de chei candidat.
4. Se calculeaza X^+ pentru fiecare dintre candidate. Daca se obtin toate atributele lui R atunci acel X este o cheie a lui R.

EURISTICA (3)

5. Se repeta pasii 3 si 4 pornind de la acele multimi candidat X care nu sunt gasite ca si chei la pasul anterior. Intre multimile candidat nu luam niciodata in considerare pe cele care contin o cheie gasita anterior.
6. Procesul se opreste cand nu se mai pot face augmentari.

EXEMPLUL 1

- R = ABCDE si
- F = { $A \rightarrow B$, $A \rightarrow C$, $D \rightarrow E$ }.
  - Multimea atributelor care nu apar in partea dreapta a nici unei dependente este X = AD.
  - Calculam $(AD)^+$. Obtinem (AD)+ = ABCDE = R.
  - Procesul se opreste. Rezulta ca AD este cheie unica pentru R

EXEMPLUL 2

- $R = ABCDE$ si
- $F = \{ A \rightarrow B, B \rightarrow A , A \rightarrow C, D \rightarrow E \}$.
  - Multimea atributelor care nu apar in partea dreapta a nici unei dependente este $X = D$.
  - Calculam $(D)^+$. Obtinem $(D)^+ = DE \neq R$. Rezulta ca D nu este cheie unica pentru R

EXEMPLUL 2 - cont

- $D^+ = DE$
  - Calculam multimea de candidate: augmentam D cu atribute din $R - D^+ = ABCDE - DE = ABC$. Obtinem AD, BD si CD
  - Calculam inchiderile lor. Obtinem $(AD)^+ = R$, $(BD)^+ = R$ si $(CD)^+ = CDE \neq R$. Rezulta ca AD si BD sunt chei ale lui R dar CD nu e cheie.

EXEMPLUL 2 – cont.

- Calculam o noua multime de candidate pornind de ca CD. Putem augmenta CD cu atribute din R – $(CD)^+$ = ABCDE – CDE = AB. Nici una dintre augmentari nu este insa posibila pentru ca atat ACD cat si BCD contin o cheie gasita anterior (AD respectiv BD).
- Procesul se opreste. Singurele chei ale lui R raman AD si BD.

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 acestul capitol va fi prezentata si forma normala 4 care se defineste in functie de alt tip de dependente, si anume dependentele multivalorice.

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 $X \to Y$ din F atunci X este supercheie pentru R

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!

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

EXEMPLUL 2

- Relatia Produse = IdP, NumeP, Qty, IdF avand asociata multimea de dependente functionale
- $F_{PRODUSE} = \pi_{PRODUSE}(F) =$
  $\{ IdP \rightarrow NumeP, IdP \rightarrow Qty, IdP \rightarrow IdF\}$
- 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)

EXEMPLUL 3

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

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 = 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 atribute prime: A, B si D.

FN3 - DEFINITIE

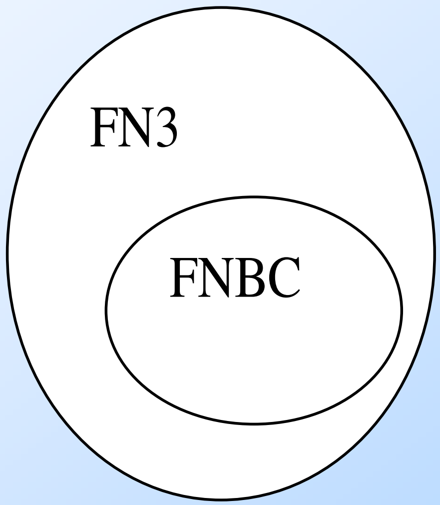
- **Definitie.** R o schema de relatie si F multimea de dependente functionale asociata. Se spune ca R este in forma normala 3 (FN3) daca si numai daca oricare ar fi o dependenta netriviala $X \rightarrow A$ din F atunci
- X este supercheie pentru R

  sau
- A este atribut prim

FN3 – cont.

- Daca in F avem dependente care contin mai multe atribute 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.

FN3 INCLUDE FNBC



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.

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

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

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.

FN1

- **Definitie:** O relatie R este in **forma normala 1** (FN1) daca pe toate atributele 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.

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).

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 atribute* din R. Atunci:
- **Definitie:** O dependenta functionala $X \to A$ se numeste ***dependenta partiala*** daca X este strict inclusa intr-o cheie a relatiei R.
- **Definitie:** O dependenta functionala $X \to A$ se numeste ***dependenta tranzitiva*** daca X nu este inclusa in nici o cheie a relatiei R.

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 F nu contine dependente partiale (dar poate contine dependente tranzitive).

EXEMPLU

- Produse = IdP, NumeP, Qty, IdF, NumeF, AdresaF

- Multimea de dependente asociata este:

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

- Este in FN2 pentru ca ultimele doua dependente sunt tranzitive dar nu sunt partiale (IdF nu apartine cheii unice IdP).

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 Sala $\rightarrow$ NrLocuri care este partiala (Sala apartine unei chei dar nu e cheie)

ATENTIE!

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

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.

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 = A1 A2 … Am .

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

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

OBSERVATII

- Schemele R_1, R_2, ..., 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)

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:

JOIN FARA PIERDERI (2)

Definitie:

- Fie R o schema de relatie,
- F multimea de dependente functionale asociata si
- o descompunere $\rho = (R_1, R_2, \ldots, R_n)$ 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 <u>care satisface dependentele F</u> avem ca:

$$r_1 \bowtie r_2 \bowtie \ldots \bowtie r_n = r \quad \text{unde}$$

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

JOIN FARA PIERDERI (3)

- 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

PASTRARE DEPENDENTE

- O a doua problema in cazul descompunerii unei scheme R avand dependentele F in mai multe relatii $R_1, R_2, \ldots, R_n$ este aceea a pastrarii corelatiilor intre date, corelatii date de dependentele functionale din F.
- Fiecare relatie R_i va mosteni o multime de dependente data de proiectia multimii de dependente functionale F pe R_i

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

EXEMPLU

- Fie relatia Produse = IdP, NumeP, Qty, IdF, NumeF, AdresaF avand dependentele functionale:
- $F = \{ IdP \rightarrow NumeP, IdP \rightarrow Qty, IdP \rightarrow IdF, IdF \rightarrow NumeF, IdF \rightarrow AdresaF \}$
- In cazul descompunerii $\rho 2 = (R_1, R_2)$ unde:
  - $R_1 = (IdP, NumeP, Qty, IdF)$
  - $R_2 = (IdF, NumeF, AdresaF)$

cele doua relatii mostenesc urmatoarele dependente:

$$F_{R1} = \pi_{R1}(F) = \{ IdP \rightarrow NumeP, IdP \rightarrow Qty, IdP \rightarrow IdF \}$$

$$F_{R2} = \pi_{R2}(F) = \{ IdF \rightarrow NumeF, IdF \rightarrow AdresaF \}$$

PASTRARE DEPENDENTE (2)

- Dupa cum se observa toate dependentele relatiei initiale sunt pastrate fie in FR1 fie in FR2.
- 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.

DEFINITIE

- Fie R o schema de relatie, F multimea de dependente functionale asociata, o descompunere $\rho = (R_1, R_2, \ldots, R_n)$ a lui R si $F_i = \pi_{R_i}(F)$ 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:

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

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

Bibliografie

1. **Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer D. Widom**: *Database Systems: The Complete Book*, Prentice-Hall, Englewood Cliffs, NJ, 2002.
2. **F. Rădulescu** : *Oracle SQL, PL/SQL*, Editura Printech, ISBN 973-718-203-02005