Navrhovanie databáz

- **Formálne metódy**:
  identifikácia, či test objektov a optimalizácia návrhu databázy pre zvolený dátový model (relačný model)
- **Poloformálne metódy**:
  analýza reálnej skutočnosti a komunikácia s koncovým užívateľom (ER-model, UML, NIAM, ...)

SQL nástroje

- Create, alter, drop
  - Domain, table, view, index, trigger
- Constraints (integritné obmedzenia)
  - Podmienky na doménu (napr.: not NULL)
  - Podmienky na tabuľku (primary key, unique, … .)
  - Medzitabuľkové podmienky (foreign key, …)

Motivácie

- Nie všetky tabuľky sú rovnako dobré
  - Študenti(meno, adresa, predmet, dátum, známka)
    Študenti(meno, adresa, idš),
    Skúšky(idš, predmet, dátum, známka)
  - Zamestnaci(meno, zaradenie, plat)
    Zamestnanci(meno, zaradenie), Mzdy(zaradenie, plat)
- Dôvody
  - Zbytočné opakovanie hodnôt (málo vadí, ale otravuje)
  - Možnosť nekonzistencie (veľmi vadí)
  - Potreba nulových hodnôt (málo vadí)
  - Anomália pri vynechávaní (veľmi vadí)

Základné pojmy pre navrhovanie v relačnom modeli

Závislosti - $\forall\exists\wedge\Rightarrow$ induktívne Hornové formuly)

Funkčné závislosti $\mathbf{x} \rightarrow \mathbf{y}$

$\forall(\mathbf{xy}_1\mathbf{z}_1\mathbf{y}_2\mathbf{z}_2)(\ R(\mathbf{xy}_1\mathbf{z}_1) \wedge R(\mathbf{xy}_2\mathbf{z}_2) \Rightarrow \mathbf{y}_1 = \mathbf{y}_2\ )$

- Multizávislosti $\mathbf{x} \twoheadrightarrow \mathbf{y}$ (multivalued dependencies)

  $(\forall\mathbf{xy}_1\mathbf{z}_1\mathbf{y}_2\mathbf{z}_2)(\ R(\mathbf{xy}_1\mathbf{z}_1)\wedge R(\mathbf{xy}_2\mathbf{z}_2) \Rightarrow R(\mathbf{xy}_1\mathbf{z}_2)\ )$

- Uzáver množiny závislostí $\boldsymbol{F}^*$ je množina všetkých závislostí, ktoré vyplývajú z $\boldsymbol{F}$.
- Úplné pokrytie je $\boldsymbol{F}^+$ je priemet $\boldsymbol{F}^*$ na závislosti daného typu
- Pokrytie množiny závislotí $\boldsymbol{F}$ je ľubovolná množina závislostí $\boldsymbol{G}$ taká, že $\boldsymbol{F}^+ = \boldsymbol{G}^+$.

Vlastnosti funkčných závislostí
(Armstrongové axiómy)

- (A1) $\mathbf{x} \subseteq \mathbf{y} \Rightarrow \mathbf{y} \rightarrow \mathbf{x}$ reflexívnosť
- (A2) $\forall \mathbf{z}\ \mathbf{x} \rightarrow \mathbf{y} \Rightarrow \mathbf{xz} \rightarrow \mathbf{yz}$ augmetation
- (A3) $(\mathbf{x} \rightarrow \mathbf{y}) \wedge (\mathbf{y} \rightarrow \mathbf{z}) \Rightarrow \mathbf{x} \rightarrow \mathbf{z}$ tranzitívnosť

Dôkaz dosadením do definície funkčnej závislosti:

(A1) $(\forall \mathbf{xyz_1z_2})(\mathbf{x} \subseteq \mathbf{y} \wedge R(\mathbf{yz_1}) \wedge R(\mathbf{yz_2}) \Rightarrow \mathbf{x} = \mathbf{x})$

(A2) $(\forall \mathbf{xzy_1y_2t_1t_2})(R(\mathbf{xzy_1t_1}) \wedge R(\mathbf{xzy_2t_2}) \Rightarrow \mathbf{y_1} = \mathbf{y_2})$
$(\forall \mathbf{xzy_1y_2t_1t_2})(R(\mathbf{xy_1zt_1}) \wedge R(\mathbf{xy_2zt_2}) \Rightarrow \mathbf{y_1z} = \mathbf{y_2z})$

(A3) $(\forall \mathbf{xy_1z_1y_2z_2t_1t_2})(R(\mathbf{xy_1z_1t_1}) \wedge R(\mathbf{xy_2z_2t_2}) \Rightarrow \mathbf{y_1} = \mathbf{y_2} = \mathbf{y})$
$(\forall \mathbf{xyz_1z_2t_1t_2})(R(\mathbf{xyz_1t_1}) \wedge R(\mathbf{xyz_2t_2}) \Rightarrow \mathbf{z_1} = \mathbf{z_2}$

Ďalšie vlastnosti funkčných závislostí

- $(\mathbf{x} \rightarrow \mathbf{y}) \wedge (\mathbf{x} \rightarrow \mathbf{z}) \Rightarrow \mathbf{x} \rightarrow \mathbf{yz}$ (union rule)
- $(\mathbf{x} \rightarrow \mathbf{y}) \wedge (\mathbf{wy} \rightarrow \mathbf{z}) \Rightarrow \mathbf{wx} \rightarrow \mathbf{wz}$ (pseudotransivity)
- $(\mathbf{x} \rightarrow \mathbf{y}) \wedge (\mathbf{z} \subseteq \mathbf{y}) \Rightarrow \mathbf{x} \rightarrow \mathbf{z}$ (decomposition)

Dôkazová technika:

$\left.\begin{array}{l} \mathbf{x} \rightarrow \mathbf{y} \Rightarrow \mathbf{x} \rightarrow \mathbf{xy} \quad \text{podľa (A2)} \\ \mathbf{x} \rightarrow \mathbf{z} \Rightarrow \mathbf{xy} \rightarrow \mathbf{yz} \quad \text{podľa (A2)} \end{array}\right\} \Rightarrow \quad \mathbf{x} \rightarrow \mathbf{yz}$ podľa (A3)

Zvyšné dôkazy sa robia podobne podľa Armstrongových axióm. (Urobte ich ako cvičenie.)

Uzáver množiny atribútov

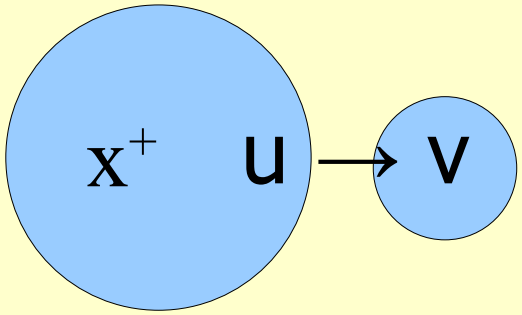
Nech **x** je množina atribútov a ***F*** je množina funkčných závislostí. Potom uzáverom $\mathbf{x}^+$ množiny **x** w.r.t. ***F*** rozumieme množinu $\mathbf{x}^+$ všetkých atribútov x takých, že $\mathbf{x} \rightarrow x$ pomocou závislostí v ***F***.

Lema: $\mathbf{x} \rightarrow \mathbf{y}$ sa dá odvodiť z ***F*** pomocou Armstrongových axióm práve vtedy keď $\mathbf{y} \subseteq \mathbf{x}^+$ w.r.t ***F***.

Pre každý atribút $a \in \mathbf{y} \subseteq \mathbf{x}^+$. Platí $\mathbf{x} \rightarrow a$ podľa definície uzáveru $\mathbf{x}^+$. Podľa union rule platí aj $\mathbf{x} \rightarrow \mathbf{y}$.

Naopak nech $\mathbf{x} \rightarrow \mathbf{y}$ sa dá odvodiť. Potom pre každe $a \in \mathbf{y}$ platí $\mathbf{x} \rightarrow a$ podľa decomposition rule a $a \in \mathbf{x}^+$.

Výpočet x^+ w.r.t. *F*



$x^+ := x;$
repeat
for each $u \rightarrow v \in F$ **do**
  if $u \in x^+$ **then** $x^+ := x^+ \cup v;$
until sa niečo pridalo;

Optimalizácia:

- Každá závislosť sa použije najviac raz, môžeme ju po použití vynechať.
- Index pre ľavé strany

Dôsledok: Uzáver x^+ sa počíta lineárne, v čase $O(n|\boldsymbol{F}|)$, kde n je počet atribútov.

Úplnosť „Armstrongových axióm“

Veta: Funkčná závislosť $\mathbf{x} \rightarrow \mathbf{y}$ sa dá odvodiť z ***F*** pomocou Armstrongových axióm práve vtedy, keď $\mathbf{x} \rightarrow \mathbf{y}$ je dôsledkom ***F***.

Dôkaz: Pretože Armstrongové axiómy sú dôsledkom definície funkčnej závislosti, dajú sa odvodiť len platné závislosti.

Opačne predpokladajme, že závislosť $\mathbf{x} \rightarrow \mathbf{y}$ platí ale nedá sa odvodiť pomocou Armstrongových axióm. Uvažujme reláciu **R**

	Atribúty x^+	ostatné atribúty
u:	1 1 … 1	1 1 … 1
v:	1 1 … 1	0 0 … 0

Všetky závislosti z ***F*** sú splnené v **R**

Nech $\mathbf{v} \rightarrow \mathbf{w}$ je dôsledkom ***F***, ale nie je splnené v **R**. Potom $\mathbf{v} \subseteq \mathbf{x}^+$. Predpoklad **w** patrí do $\mathbf{x}^+$ aj **w** nepatrí do $\mathbf{x}^+$ vedie k sporu.

Charaktrizácia úplného pokrytia množiny funkčných závislostí

Množina F^+ je príliš obsiahla, stačí však uvádzať maximálne závislosti. Závislosť je maximálna ak nemôžeme vynechať žiaden atribút na ľavej strane alebo pridať nejaký atribút na pravú stranu bez porušenia jej platnosti.

K charakterizácii stačia nasýtené množiny = pravé strany maximálnych závislostí. (Spätná rekonštrukcia maximálnych závislostí.)

Veta: Každá úplná množina funkčných závislostí má model v nejakej relácii nad doménou D= { 0, 1 }.

Vyplývanie medzi funkčnými závislosťami

Výpočet F^+ a testovanie ekvivalencie $F^+ = G^+$ je vo všeobecnosti náročná (exponenciálna) záležitosť. Našťastie stačí počítať uzávery množiny atribútov vzhľadom k F^+.

Hovoríme aj, že G pokrýva F.

Minimálne pokrytie množiny funkčných závislostí

Kánonické závislosti na pravej strane len jeden atribút.

Minimálne pokrytie je pokrytie kánonickými závislosťami, z ktorých sa žiadna nedá vynechať bez toho, aby sa porušila vlastnosť byť pokrytím.

$AB \rightarrow C$
$CG \rightarrow D$
$CE \rightarrow A$
$CE \rightarrow G$

$D \rightarrow E$
$C \rightarrow A$
$BC \rightarrow D$
$ACD \rightarrow B$

$CG \rightarrow B$
$D \rightarrow G$
$BE \rightarrow C$

Minimálne pokrytia

$AB \rightarrow C$
$C \rightarrow A$
$BC \rightarrow D$
$D \rightarrow E$
$D \rightarrow G$
$BE \rightarrow C$
$CE \rightarrow G$
$CD \rightarrow B$
$CG \rightarrow D$

$AB \rightarrow C$
$C \rightarrow A$
$BC \rightarrow D$
$D \rightarrow E$
$D \rightarrow G$
$BE \rightarrow C$
$CE \rightarrow G$
$CG \rightarrow B$

Nadkľúče a kľúče

Nech je daná relácia **R**(**U**). Potom množinu atribútov ***K*** takú, že $\boldsymbol{K} \rightarrow \boldsymbol{U}$ nazývame nadklúč. Minimálny nadklúč v zmysle množinovej inklúzie nazývame klúč.

Koľko kľúčov môže mať relácia o n atribútoch ?

Príklad:

R(A_1, ..., A_k, B_1, ..., B_k, C)

$\boldsymbol{F} = \{ A_i \leftrightarrow B_i \text{ pre } 1 \leqslant i \leqslant k \} \cup \{ A_1 \ldots A_k \rightarrow C \}$

Bezstrátové spojenia

$\rho=\{R_1, \ldots, R_k\}$, $R = R_1 \cup \ldots \cup R_k$

$$m_\rho(r) = \Pi_{R_1}(r) \bowtie \ldots \bowtie \Pi_{R_k}(r)$$ Join project mapping

Vlastnosti: $r \subseteq m_\rho(r)$

$$m_\rho(r) = m_\rho(m_\rho(r))$$

Hovoríme, že dekompozícia má bezstrátové spojenie ak $r = m_\rho(r)$. Tiež, že v R plati spojovacia závislosť (JD)

Tabuľková metóda testovania.

R= SAIP

$S \rightarrow A$

$SI \rightarrow P$

S	A	I	P
a_1	a_2	b_{13}	b_{14}
a_1	b_{23}	a_3	a_4

Normálne formy (BCNF, 3NF)

BCNF: Relačná schéma R je v BCNF, keď pre každú v nej platnú funkčnú závislosť **x** $\rightarrow$ **y** platí **x** je nadkľúč.

3NF: Relačná schéma R je v 3NF, keď pre každú v nej platnú funkčnú závislosť **x** $\rightarrow$ **y** platí **x** je nadkľúč alebo **y** je prvok nejakého kľúča (primárny atribút) **R**.

Lema: a.) Každá binárna relácia je v BCNF.
b.) Ak R nie je v BCNF. Potom v nej existujú atribúty A a B také, že (**R** - AB) $\rightarrow$ A.
(Môže a nemusí platiť (**R** - AB) $\rightarrow$ B.)

Naivná dekompozícia do _NF

- Najdi závislosť z F^+, ktorá porušuje podmienku _NF. NP – úplná úloha !
- Minimalizuj jej ľavú stranu.
- Maximalizuj jej pravú stranu (nie je nutné).
- Nech výsledok je $\mathbf{x} \rightarrow \mathbf{y}$.
- Vytvor dve nové relácie R – **y** a **xy**.
- S vzniklými reláciami proces opakuj, pokiaľ všetky nie sú v požadovanej _NF.

Celé to funguje lebo **x** nemôže byť nadkľúč, závislosť by neporušovala podmienku _NF. Počet atribútov v reláciach sa teda zmenšuje.

3NF zachovávajúca závislosti

Príklad: R= MAP (Mesto, Adresa, PSČ)
Závislosti: MA $\rightarrow$ P, P $\rightarrow$ M

Hovoríme že dekompozícia **R** = $\mathbf{R}_1 \cup \ldots \cup \mathbf{R}_k$ zachováva závislosti ***F*** , ak každá závislosť z ***F*** je v uzávere takých závislostí **x** $\rightarrow$ **y** z ***F***, že **xy** $\subseteq \mathbf{R}_i$.

Algoritmus testovania zachovania závislosti **x** $\rightarrow$ **y**

z := **x**; **while** sa **z** zmenilo **do**

for i := 1 **to** k **do**

z := **z** $\cup$ ((**z** $\cap$ $\mathbf{R}_i$)$^+$ $\cap$ $\mathbf{R}_i$) {uzáver w.r.t. ***F*** };

Algoritmus normalizácie do BCNF

Vstup: Relačná schéma **R** a množina funkčných závislostí ***F***.

Výstup: Množina relačných schém R_1 ... R_k v BCNF.

Metóda: Dekompozícia na dve schémy podľa predošlej lemy jednu **X**A zodpovedajúcu závislosti **X** $\rightarrow$ A, ktorá je v BCNF a druhú **R** - A, na ktorú použijeme algoritmus rekurzívne.

Algoritmus normalizácie podrobnejšie

```
Z :=  R;
repeat  bcnf:= decompose( Z, Y, A);
            Z :=  Z - A;
until bcnf;
```

```
function decompose(in Z, out Y, out A): boolean ;
{ if Z neobsahuje atribúty A, B také, že A ∈ (Z - AB)+ then
        begin  Y:=  Z; bcnf= true end
   else  begin najdi A a B;
            Y:= Z - B;
            while Y obsahuje A a B také, že A ∈  (Y - AB)+ do
                     Y:= Y - B;
            bcnf := false;
         end;
   return(bcnf) }
```

Normalizácia do 3NF zachovávajúcej závislosti

Vstup: Relačná schéma **R** a minimálne pokrytie ***F.***

Výstup: Relačné schémy bestrátovej dekompozície do 3NF.

Metóda: Ak ***F*** obsahuje závislosť, ktorá obsahuje všetky atribúty **R**, potom **R** je už v 3NF.
Inak každej funkčnej závislosti v ***F*** zodpovedá jedna relačná schéma. Treba pridať ešte relačnú schému pre atribúty **R**, ktoré sa nevyskytujú v žiadnej funkčnej závislosti ***F***. Tieto atribúty musia byť súčasťou každého kľúča, aby došlo k spojeniu treba ich doplniť na kľúč **R**.

Pravidlá pre multizávislosti

- (A1) $\mathbf{x} \subseteq \mathbf{y} \subseteq \mathbf{U} \Rightarrow \mathbf{y} \rightarrow \mathbf{x}$ reflexívnosť
- (A2) $\forall \mathbf{z}\ \mathbf{x} \rightarrow \mathbf{y} \Rightarrow \mathbf{xz} \rightarrow \mathbf{yz}$ augmentation
- (A3) $(\mathbf{x} \rightarrow \mathbf{y}) \wedge (\mathbf{y} \rightarrow \mathbf{z}) \Rightarrow \mathbf{x} \rightarrow \mathbf{z}$ tranzitívnosť
- (A4) $\mathbf{x} \twoheadrightarrow \mathbf{y} \Rightarrow \mathbf{x} \twoheadrightarrow \mathbf{U} - \mathbf{x} - \mathbf{y}$ complementation
- (A5) $(\mathbf{x} \twoheadrightarrow \mathbf{y}) \wedge (\mathbf{v} \subseteq \mathbf{w}) \Rightarrow \mathbf{wx} \twoheadrightarrow \mathbf{vy}$ augmentation
- (A6) $(\mathbf{x} \twoheadrightarrow \mathbf{y}) \wedge (\mathbf{y} \twoheadrightarrow \mathbf{z}) \Rightarrow \mathbf{x} \twoheadrightarrow (\mathbf{z} - \mathbf{y})$
- (A7) $\mathbf{x} \rightarrow \mathbf{y} \Rightarrow \mathbf{x} \twoheadrightarrow \mathbf{y}$
- (A8) $(\mathbf{x} \twoheadrightarrow \mathbf{y}) \wedge (\mathbf{z} \subseteq \mathbf{y}) \wedge (\mathbf{w} \cap \mathbf{y} = \emptyset) \wedge (\mathbf{w} \rightarrow \mathbf{z}) \Rightarrow \mathbf{x} \rightarrow \mathbf{z}$

4NF

V relačnej schéme **R** je multizávislosť $\mathbf{x} \twoheadrightarrow \mathbf{y}$ triviálna, ak $\mathbf{y} \subseteq \mathbf{x}$ alebo $\mathbf{x} \cup \mathbf{y} = \mathbf{R}$.

Nech $\mathbf{D}^+$ je množina všetkých platných závislostí a multizávislostí v relačnej schéme **R**. Hovoríme, že relačná schéma **R** je v **4NF**, ak pre každú netriviálnu multizávislosť $\mathbf{x} \twoheadrightarrow \mathbf{y}$ platí, že **x** je nadkľúč.

Veta: $4NF \Rightarrow BCNF \Rightarrow 3NF$

5NF

Spojovacia závislosť : $R = \bowtie_{i=1}^{k} \Pi_{R_i}$

Spojovacia závislosť je triviálna, ak pre nejaké i $R_i = R$

Definícia 5NF

Relácia R je v 5NF, ak pre každú netriviálnu spojovaciu závislosť v R platí: všetky R_i sú nadkľúče v R.

Ak sa niečo „kartézuje“, alebo „skoro kartézuje“ je lepšie mať to v samostatných tabuľkách.

Inklúzne závislosti

Inklúzne závislosti sa najjednoduchšie definujú v algebre. Nech R_1 a R_2 sú dve relácie so spoločnou množinou atribútov **x**. Podmienku $\Pi_{\mathbf{x}}R_2 \subseteq \Pi_{\mathbf{x}}R_1$ nazývame inklúznou závislosťou. $\forall \mathbf{x}(\exists \mathbf{y} R_2(\mathbf{x},\mathbf{y}) \Rightarrow \exists \mathbf{z} R_1(\mathbf{x},\mathbf{z}))$

Príklad:
Študenti(sid, meno, priezvisko, rokzap, …)
Známky(sid, predmet, známka)
Byrokracia: Známku možno zapísať len študentovi.
sid je v tabuľke známky cudzí kľúč (foreign key).

Špeciálna podpora inklúznych závislostí v SQL referenčná integrita (kaskádovité vkladanie a vynechanie).

Zložitosť odvodzovania dôsledkov

- Len funkčné závislosti – polynomiálna
- Multizávislosti a spojovacie závislosti (môžeme pridať aj funkčné závislosti), používa sa uzatváracia procedúra (chase) – exponenciálna zložitosť
- Ak pridáme inklúzne závislosti – nerozhodnuteľné

Názor praktikov

Are Normal Forms a good thing ?

NOT obvious !

1. Decomposition may lead to poor performance
   get the semantics right first, then tune the performance by caching or whatever
   self-maintaining materialised views ?
2. Automated decomposition may generate unnatural database designs
   large-scale design will require tools, but the schema generated may need tuning
3. Decomposition may break referential integrity
   use the ***FOREIGN KEY*** directive of ***SQL*** to enforce the links via the schema

Umiernený pohľad na normalizáciu

- Databáza nemusí byť v nNF
- Nemala by sa však priveľmi líšiť od normalizovanej schémy
- Ak niektoré tabuľky nie sú v nNF, návrhár, či správca databázy, by mali poznať „technické dôvody“, prečo to tak je.