

---

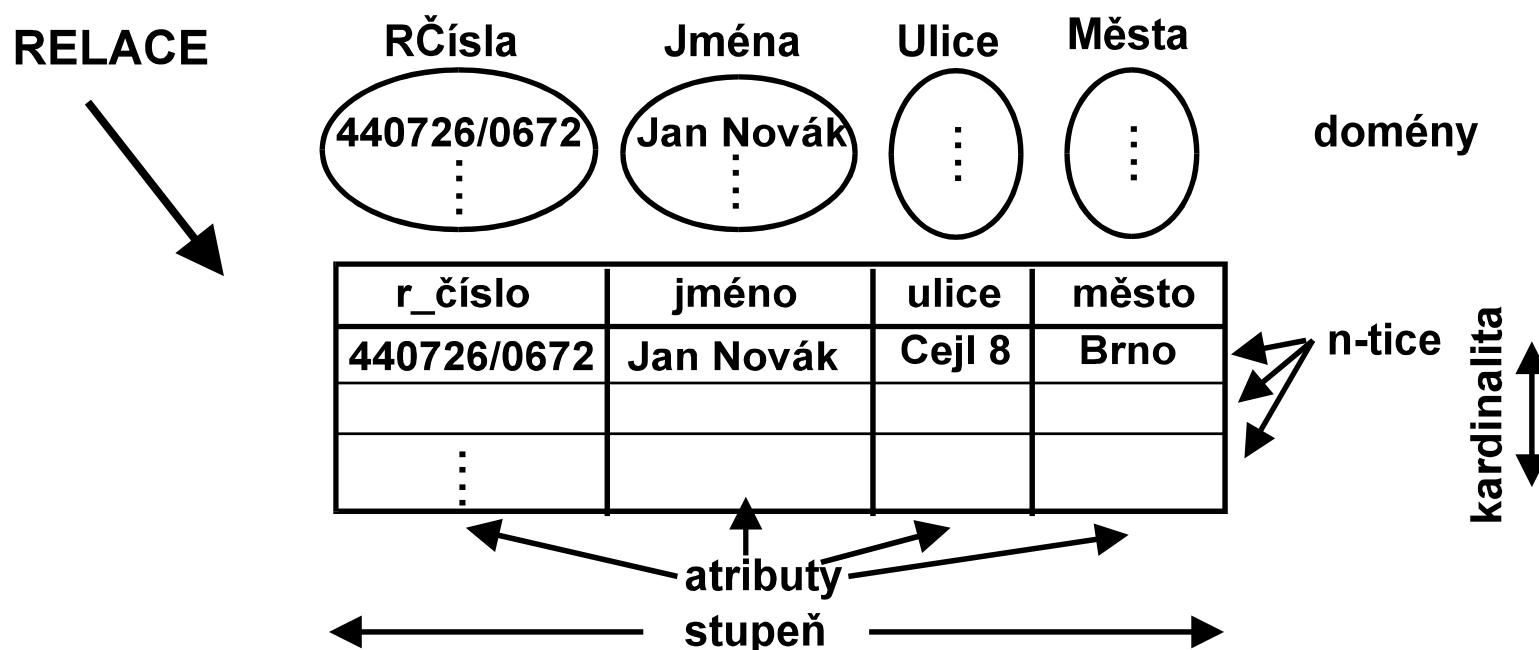
# 4. Relační model dat

4.1. Relační struktura dat.....	3
4.2. Integritní pravidla v relačním modelu .....	9
4.2.1. Primární klíč.....	9
4.2.2. Cizí klíč.....	11
4.2.3. Relační schéma databáze.....	13
4.3. Relační algebra .....	15
4.4. Relační kalkuly.....	22
4.4.1. N-ticový relační kalkuly (NRK) .....	23
4.4.2. Doménový relační kalkuly (DRK).....	25
Literatura.....	26

## **1970 - E.F.Codd: „A relational data model for large shared data banks“**

- **Přínos publikace:**
  - **Oddělení logické struktury dat od implementace**
  - **Transparentnost přístupových metod při manipulacích s daty,**
  - **Poskytnutí matematické podpory pro manipulaci s daty**
  - **Poskytnutí matematické podpory k omezení redundance při návrhu logické struktury databáze**
- **Složky relačního modelu dat**
  - **Relační datová struktura**
  - **Obecná integritní omezení pro relační databáze**
  - **Manipulace s daty v relační databázi**

## 4.1. Relační struktura dat



**Doména** - pojmenovaná množina skalárních hodnot téhož typu.

Př) Doména názvů měst

**Skalární hodnota** - nejmenší sémantická jednotka dat, atomická (vnitřně nestrukturovaná).

Př) Josef Novák

**Složená doména** – doména složená z několika jednoduchých domén.

**Př) (Josef, Novák)**

- Každý atribut je definován na nějaké doméně -  $(A_i:D_i)$ . Jednoduchý atribut na jednoduché, složený na složené.
- Domény omezují porovnávání hodnot.

*Poznámka: Většina SŘBD pojem domén nepodporuje vůbec nebo jen částečně (pojmenované standardní datové typy).*

*Př) SQL/92, ne Oracle, SQLBase*

```
CREATE DOMAIN Města CHAR[20] DEFAULT '???'  
CREATE TABLE Zákazník (... , město Města, ...)
```

- Relace

**Relace** na doménách  $D_1, D_2, \dots, D_n$  je dvojice  $R = (R, R^*)$ , kde  $R = R(A_1:D_1, A_2:D_2, \dots, A_n:D_n)$  je **schéma relace** a  $R^* \subseteq D_1 \times D_2 \times \dots \times D_n$  je tělo relace. Schéma relace zapisujeme často zjednodušeně ve tvaru  $R(A_1, A_2, \dots, A_n)$ . Počet atributů  $n$  relace se označuje **stupeň (řád) relace**, kardinalita těla relace  $m = |R^*|$  se označuje **kardinalita relace**.

- Stupeň relace je konstantní, kardinalita proměnná

- Vztah pojmu „relace“ v relačním modelu dat a pojmu „tabulka“

<b>relace</b>	základní abstraktní pojem relačního modelu
<b>tabulka</b>	forma znázornění relace

**Poznámka:** Název „relační model“ a „relační databáze“ je odvozen od faktu, že relace je základním abstraktním pojmem modelu a jedinou strukturou databáze na logické úrovni.

- Alternativní definice – *záhlaví* + *tělo* relace

**Relace**  $R$  na doménách  $D_1, D_2, \dots, D_n$  je dvojice  $R = (H, B)$ , kde  $H$  značí záhlaví relace a  $B$  tělo relace.

**Záhlaví relace** je množina:

$H = \{(A_1:D_1), (A_2:D_2), \dots, (A_n:D_n)\}$   $A_i \neq A_j$  pro  $i \neq j$ ,  
 $A_i$  ( $i = 1, \dots, n$ ) značí atributy a  $D_i$  ( $i=1, \dots, n$ ) jsou odpovídající domény.

**Tělo relace** je tvořeno časově proměnnou množinou  $n$ -tic:

$B(t) = \{r_1, r_2, \dots, r_m(t)\}$ , kde  $r_i = \{(A_1:v_{i1}), (A_2:v_{i2}), \dots, (A_n:v_{in})\}$   
 $i = 1, 2, \dots, m(t)$ ,  $n$  - stupeň (řád) relace,  $m$ - kardinalita relace.

- Vlastnosti relace:

- Neexistují duplicitní  $n$ -tice,
- $n$ -tice jsou neuspořádané
- Hodnoty jednoduchých atributů jsou atomické

- „Prohřešky“ SQL

- DISTINCT/ALL(?), NEXT (?)

- Přínos požadavku normalizované relace (v 1NF)
  - Normalizovaná relace je jednodušší, stačí jednodušší operace.

## Př) relace Účet

### Nenormalizovaná

r_ číslo	účet	
	č_ účtu	pobočka
4440726/0672	4320286	Jánská
	2075752	Palackého
530610/4532	1182648	Palackého

### Normalizovaná

r_ číslo	č_ účtu	pobočka
4440726/0672	4320286	Jánská
4440726/0672	2075752	Palackého
530610/4532	1182648	Palackého

- **Typy relací**
  - **pojmenované**
    - **bázové (reálné)**
    - **pohledy (virtuální)**
    - **materializované pohledy (snapshot)- odvozené, ale existující**
    - **dočasné**
  - **nepojmenované**
    - **výsledky dotazů**
    - **mezivýsledky**

### **Př) SQL/92 - příklad přechodné tabulky pro SQL sezení**

```
CREATE GLOBAL TEMPORARY TABLE Temp (...)  
    [ON COMMIT {PRESERVE | DELETE} ROWS]
```

- **Shrnutí základních vlastností relační struktury**
  - **Relační databáze je vnímána uživatelem (aplikací) jako kolekce časově proměnných normalizovaných relací.**
  - **Veškerá data v relační databázi jsou reprezentována explicitní hodnotou (žádné ukazatele apod.).**

## 4.2. Integritní pravidla v relačním modelu

- omezení plynoucí z reality reprezentované daty v databázi
- Typy integritních omezení:
  - specifická – pro konkrétní aplikaci
  - obecná – musí platit v každé databázi daného typu
- obecná omezení v relačním modelu se týkají primárních a cizích klíčů

### 4.2.1. Primární klíč

- atribut, který jednoznačně identifikuje n-tici v relaci

Atribut  $k$  relace  $R$  se nazývá **kandidátním klíčem**, když má tyto dvě časově nezávislé vlastnosti:

1. jednoznačnost
2. minimalita (neredukovatelnost).

- relaci lze chápat jako paměť s asociativním výběrem
- každá relace má alespoň jeden kandidátní klíč
- atribut, který je součástí kandidátního klíče budeme nazývat **klíčový**

**Primárním klíčem** je jeden z kandidátních klíčů (vybraný), zbývající kandidátní klíče se nazývají **alternativní** (někdy také **sekundární**).

- způsob výběru primárního klíče není v relačním modelu specifikován
- existence primárního klíče neimplikuje existenci odpovídajícího indexu
- primární klíč je základním prostředkem adresace n-tic v relačním modelu

### Př) Použití hodnoty primárního klíče pro adresaci

```
SELECT *  
FROM Klient  
WHERE r_cislo='440726/0672'
```

```
SELECT *  
FROM Klient  
WHERE jmeno='Jan Novák';
```

## Pravidlo integrity entit

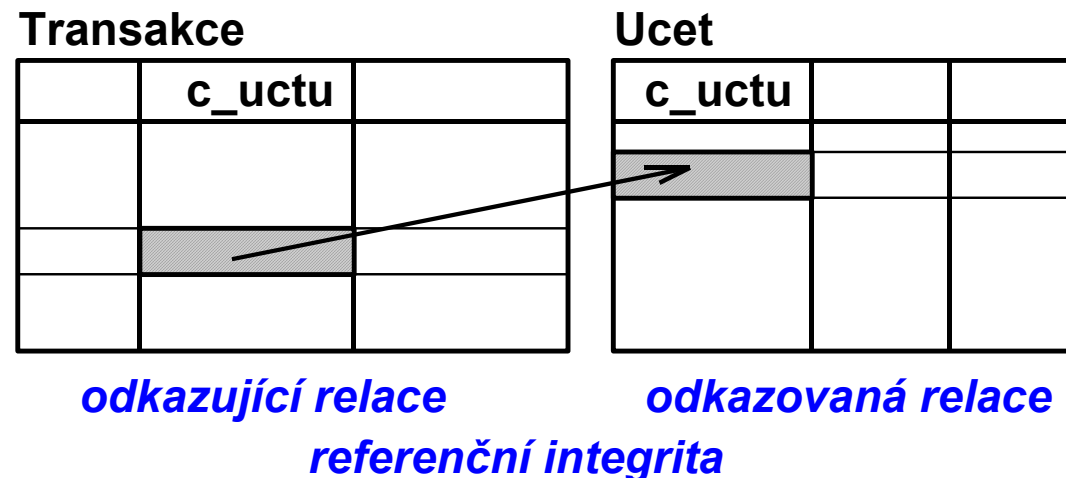
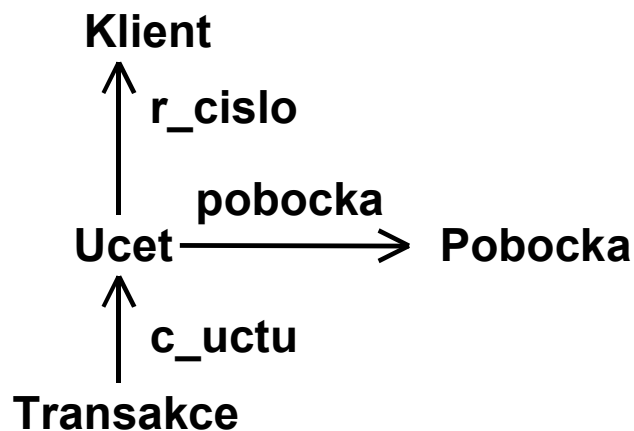
U žádné komponenty primárního klíče báze relace nesmí chybět hodnota (nesmí být NULL).

- entity jsou identifikovatelné → n-tice musí být také
- pravidlo se vztahuje na báze relace
- pravidlo se týká pouze primárního klíče

⇒ Každá n-tice báze relace musí být v každém okamžiku jednoznačně identifikovatelná hodnotou primárního klíče.

### 4.2.2. Cizí klíč

Př) transakce.č\_úctu - je hodnota 5270817 legální?



Atribut *FK* báze relace **R2** se nazývá *cizí klíč*, právě když splňuje tyto časově nezávislé vlastnosti:

1. Každá hodnota *FK* je buď plně zadaná nebo plně nezadaná.
  2. Existuje báze relace **R1** s primárním klíčem *PK* takovým, že každá zadaná hodnota *FK* je identická s hodnotou *PK* nějaké *n*-tice relace **R1**.
- cizí a odpovídající primární klíč by měly být definovány na téže doméně
  - referenční cesta, sebeodkazující relace, referenční cykly
  - soulad hodnot cizích a primárních klíčů představuje vztahy mezi *n*-ticemi ("drží databázi pohromadě")

### *Pravidlo referenční integrity*

DB nesmí obsahovat žádnou nesouhlasnou hodnotu cizího klíče.

- pravidlo referenční integrity se týká stavu databáze

### 4.2.3. Relační schéma databáze

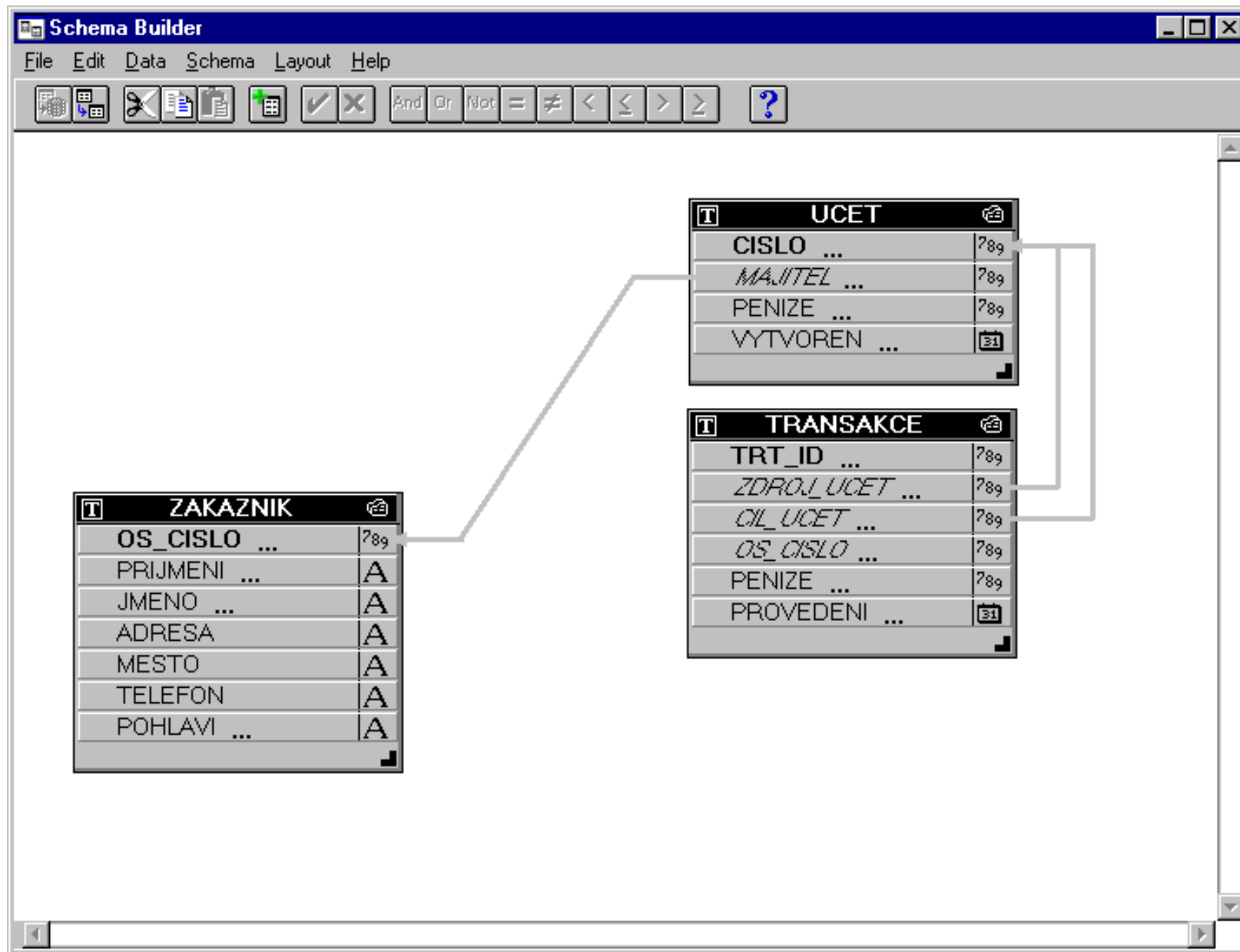
**Relačním schématem databáze** nazýváme dvojici  $(R, I)$ , kde  $R = \{R_1, R_2, \dots, R_k\}$  je množina schémat relací a  $I = \{I_1, I_2, \dots, I_l\}$  je množina integritních omezení.

Někdy jsou lokální integritní omezení rozdělena mezi jednotlivá schémata, tj.

$$R = \{(R_1, I_1), (R_2, I_2), \dots, (R_k, I_k)\}$$

**(Přípustnou) relační databází** se schématem  $(R, I)$  nazýváme množinu relací  $R_1, R_2, \dots, R_k$  takových, že prvky těla relací splňují všechna integritní omezení z  $I$ . Říkáme také, že relace, resp. data jsou konzistentní.

- Diagram schématu databáze



## 4.3. Relační algebra

**Relační algebrou** rozumíme dvojici  $RA = (R, O)$ , kde nosičem  $R$  je množina relací a  $O$  je množina operací, která zahrnuje:

- tradiční množinové operace (sjednocení, průnik, rozdíl, součin),
- speciální relační operace, mezi které patří projekce, selekce (restrikce), spojení a dělení.

- **Tradiční operace**

Relace jsou množiny  $n$ -tic, proto mají tradiční operace obvyklý význam s respektováním vlastností relací (není libovolná množina).

**Sjednocením** relací  $R1 = (R, R1^*)$  a  $R2 = (R, R2^*)$  se schématem  $R$  je relace  $R1 \text{ UNION } R2 = (R, R1^* \cup R2^*)$ .

Analogicky pro **průnik** ( $R1 \text{ INTERSECT } R2$ ) a **rozdíl** ( $R1 \text{ MINUS } R2$ ).

**Kartézským součinem** relací  $R1 = (R1, R1^*)$  a  $R2 = (R2, R2^*)$  je relace  $R1 \text{ TIMES } R2 = ((R1, R2), R1^* \times R2^*)$ .

- Speciální relační operace

- projekce, selekce (restrikce) - unární; spojení, dělení – binární

- Projekce

**Projekce** relace **R1** na atributy **X, Y, ..., Z** ( je relace

**R1[X, Y, ..., Z]**

se schématem **(X, Y, ..., Z)** a tělem zahrnujícím všechny n-tice

**t = (x, y, ..., z)** takové, že v **R1\*** existuje n-tice **t'** s hodnotou atributu **X** rovnou **x**, **Y** rovnou **y**, ... **Z** rovnou **z**.

**Př) Klient [r\_číslo, jméno]**

r_číslo	jméno
440726/0672	Jan Novák
530610/4532	Petr Veselý
601001/2218	Ivan Zeman
510230/048	Pavel Tomek
580807/9638	Josef Mádr
625622/6249	Jana Malá

## ➤ Selekce (restrikce)

Nechť  $\theta$  reprezentuje operátor porovnání dvou hodnot ( $<$ ,  $>$ ,  $<>$ ,  $=$ , atd.).  $\theta$  **selekce (restrikce)** relace  $R$  na attributech  $X$  a  $Y$  je relace

**$R \text{ WHERE } X \theta Y$ ,**

kteřá má stejné schéma jako relace  $R$  a obsahuje všechny  $n$ -tice  $t \in R^*$ , pro které platí  $x \theta y$ , kde  $x$  je hodnota atributu  $X$  a  $y$  hodnota atributu  $Y$  v  $n$ -tici  $t$ .

- na místě buď  $X$  nebo  $Y$  může být literál

**Př) Klient where r\_číslo = '440726/0672'**

r_číslo	jméno	ulice	město
440726/0672	Jan Novák	Cejl 8	Brno

- rozšíření podmínky o logické spojky:

**$R1 \text{ WHERE } c1 \text{ AND } c2 \equiv (R1 \text{ WHERE } c1) \text{ INTERSECT } (R1 \text{ WHERE } c2)$**

- podobně OR, NOT.

## ➤ Spojení

Nechť **R1** je relace se schématem  $R1(X1, X2, \dots, X_m, Y1, Y2, \dots, Y_n)$  a **R2** relace se schématem  $(Y1, Y2, \dots, Y_n, Z1, Z2, \dots, Z_k)$ .

Uvažujme složené atributy  $X=(X1, X2, \dots, X_m)$ ,  $Y=(Y1, Y2, \dots, Y_n)$  a  $Z=(Z1, Z2, \dots, Z_k)$ . Potom **přirozené spojení** relací **R1** a **R2** je relace

### **R1 JOIN R2**

se schématem  $(X, Y, Z)$  a tělem zahrnujícím všechny  $n$ -tice  $t = (x, y, z)$  takové, že v  $R1^*$  existuje  $n$ -tice  $t'$  s hodnotou  $x$  atributu  $X$  a hodnotou  $y$  atributu  $Y$  a v  $R2^*$  existuje  $n$ -tice  $t''$  s hodnotou  $y$  atributu  $Y$  a hodnotou  $z$  atributu  $Z$ .

## Př) Účet JOIN Transakce

č_úctu	stav	r_číslo	č_úctu	č_transakce
4320286	52000	440726/0672	4320286	1
1182648	10853	530610/4532	4320286	2

č_úctu	stav	r_číslo	č_transakce
4320286	52000	440726/0672	1
4320286	52000	440726/0672	2

- Další typy spojení

Obecné spojení - *theta join*:

$R1 [X \theta Y] =df (R1 \text{ TIMES } R2 \text{ WHERE } X \theta Y)$

*Polospojení* (levé):  $R1 \text{ SEMIJOIN } R2 =df (R1 \text{ JOIN } R2) [atributy R1]$

*Vnější spojení* (levé):  $R1 \text{ OUTER JOIN } R2$

➤ Dělení

Nechť  $R1$  je relace se schématem  $(X, Y)$ , kde  $X$  a  $Y$  jsou obecně složené atributy a  $R2$  relace se schématem  $(Y)$ . Výsledkem *dělení* relace  $R1$  relací  $R2$  je relace

$R1 \text{ DIVIDEBY } R2$

se schématem  $(X)$  a tělem obsahujícím množinu všech  $n$ -tic  $t = (x)$  takových, že v  $R1^*$  existují  $n$ -tice  $t_1 = (x, y_1), t_2 = (x, y_2), \dots, t_k = (x, y_k)$ , kde  $y_1, y_2, \dots, y_k$  jsou všechny hodnoty atributu  $Y$  v  $n$ -ticích v  $R2^*$ .

**Př) Účet [r\_číslo, pobočka] DIVIDEBY Pobočka[RENAME název TO pobočka]**

r_číslo	pobočka
440726/0672	Jánská
530610/4532	Palackého
440726/0672	Palackého

název
Jánská
Palackého

r_číslo
440726/0672

- **Minimální množina operací relační algebry**
  - Sjednocení, rozdíl, kartézský součin, projekce, selekce
- **Rozšířená relační algebra**
  - Definice dalších operací a pojmů (přiřazení, přejmenování (rename), agregační funkce,...)

*Poznámka: V literatuře se často používají pro operace relační algebry následující symboly:*

$\sigma_{\theta} (R)$	$R \text{ WHERE } \theta$
$\Pi_{X, Y}(R)$	$R [X, Y]$
$R \bowtie S$	$R \text{ JOIN } S$
$R \div S$	$R \text{ DIVIDEBY } S$

- Výrazy relační algebry jako dotazovací jazyk

Jazyk výrazů relační algebry je procedurálním dotazovacím jazykem.

Př) „Kteří klienti prováděli transakce v říjnu?“

(Klient JOIN Účet JOIN Transakce) WHERE datum >= '1.10.2003'  
AND datum <= '31.10.2003'

- Význam relační algebry

- vhodný základ pro optimalizaci zpracování dotazů
- referenční prostředek pro hodnocení vlastností a porovnání relačních dotazovacích jazyků

Databázový jazyk je *relačně úplný* (relationally complete), je-li alespoň tak mocný jako relační algebra.

## 4.4. Relační kalkul

- dotazovací jazyk na bázi logiky
- neprocedurální dotazovací jazyk (CO chceme dostat)

Př) „Kteří klienti mají účet u pobočky Jánská?“

RA:

(Klient JOIN Účet) WHERE pobočka='Jánská'

RK (zavedený Coddem):

$\{k | k \in \text{Klient} \wedge \exists u \in \text{Účet} (k.r\_číslo = u.r\_číslo \wedge u.pobočka = 'Jánská') \}$

- Typy relačního kalkulu
  - n-ticový relační kalkul (NRK)
  - doménový relační kalkul (DRK)

### 4.4.1. N-ticový relační kalkulus

- **Termy**
  - *n-ticové proměnné* - definované na *n-ticích* relacích
  - odkazy na atributy (např. z.jméno)
  - konstanty (hodnoty z domény)
- **Predikátové symboly** -  $\{<, >, <=, >=, =, <>\} \dots \theta$
- **Atomická formul**
  - $R(x)$ , kde  $R$  je relace a  $x$  je *n-ticová* proměnná
  - $x.A \theta y.B$ , resp.  $x.A \theta 'c'$
- **Formule (WFF – Well Formed Formula)**
  - atomická formule
  - je-li  $P1$  formule, pak i  $NOT P1$  a  $(P1)$  jsou formule
  - jsou-li  $P1$  a  $P2$  formule, pak i  $P1 AND P2$ ,  $P1 OR P2$  a  $IF P1 THEN P2$  jsou formule
  - je-li  $P(x)$  formule s volnou *n-ticovou* proměnnou  $x$ , pak i  $EXISTS x \in R (P(x))$  a  $FORALL x \in R (P(x))$  jsou formule (s tzv. omezenými kvantifikátory).

**Jazyk NRK** jsou výrazy tvaru:

***seznam\_komponent\_volných\_proměnných where formule\_dotazu***

- **Problém bezpečnosti výrazů**

Výraz je *bezpečný*, pokud všechny hodnoty výsledku jsou vytvořeny z n-tic v databázi.

Př) výraz, který není bezpečný

**x WHERE NOT R(x)**

- pro bezpečné výrazy musí být syntaxe omezena

**NRK omezený na bezpečné výrazy je ekvivalentní relační algebře.**

Př) "Kteří klienti mají účet u pobočky Jánská?"

**k WHERE Klient(k) AND EXISTS u ∈ Účet(u.r\_číslo = k.r\_číslo AND u.pobočka = 'Jánská')**

## 4.4.2. Doménový relační kalkul (DRK)

- Termy
  - doménové proměnné - definované na doménách
  - konstanty (hodnoty z domény)
- Predikátové symboly -  $\{<, >, <=, >=, =, <>\} \dots \theta$
- Atomická formule:  
 $R(A_1:t_1, A_2:t_2, \dots, A_n:t_n),$   
 $t_1 \theta t_1$
- Formule - viz NRK
- Problém bezpečnosti výrazů
  - analogicky jako u NRK

Př) výraz, který není bezpečný

*x where not R(jméno:x)*

Př) "Kteří klienti mají účet u pobočky Jánská?"

*rc, j, u, m WHERE Klient(r\_číslo: rc, jméno: j, ulice: u, město: m)  
AND Účet(r\_číslo:rc, pobočka: 'Jánská')*

## Literatura

1. **Silberschatz, A., Korth H.F, Sudarshan, S.: Database System Concepts. Fourth Edition. McGRAW-HILL. 2001, str. 79 – 131.**
2. **Pokorný, J.: Dotazovací jazyky. Science, Veletiny, 1994, str. 21 – 46.**
3. **Pokorný, J.: Databazová abeceda. Science, Veletiny, 1998, str. 145 – 148.**
4. **Date C.J.: An Introduction to Database Systems. Sixth edition. Addison-Wesley, 1995, str. 79 – 218.**