

DBS – BI-SPOL-9

Relační databáze, dotazování v relační algebře, základní koncepce jazyka SQL (SELECT, DDL, DML, DCL, TCL) , vyjádření integritních omezení v DDL.

Obsah

1	DBS	2
2	Relační databáze	2
3	Relační algebra	2
3.1	Základní operace relační algebry	2
4	Polospojení	3
5	SQL	4
6	Integritní omezení	4

1 DBS

DBS = database system

DBMS = systém řízení bází dat (database management system)

- RDBMS = relační DBMS
- ODBMS = objektový DBMS
- ORDBMS = objektově-relační DBMS

Zabývá se seřazením velkého množství, perzistentních, spolehlivých a sdílených dat

- velké množství = pro data nestačí operační paměť
- perzistentní = data přetrvávají od zpracování ke zpracování
- spolehlivý = data lze rekonstruovat po chybě
- sdílených = data jsou přístupná více uživatelům

2 Relaçní databáze

(R,I) je schéma relační databáze, kde

- $R = R_1, R_2, \dots, R_n$ je množina relací
- I je množina integritních omezení

relace = množina n-tic $\subset D_1 \times D_2 \times \dots \times D_n$ (relace = tabulka)

- jména atributů $[A_1, A_2, \dots, A_n]$
- domény atributů D_i
- v relaci nezáleží na pořadí n-tic
- relace neobsahuje duplicitní n-tice

3 Relaçní algebra

- relační algebra je "vyšší" jazyk
- nespecifikujeme "jak se mají věci dělat", ale "co má být výsledkem"
- výsledek dotazu je relace, která může být vstupem do dalšího dotazu - jdou řetěžit
- řeší "pouze" dotazování nikoliv DML ani DDL

3.1 Základní operace relační algebry

- selekce (restrikce) = relace R dle podmínky ϕ
 - $R(\phi = \text{def}\{u|u \in R \wedge \phi(u)\})$ = množina splňující podmínku
- projekce = relace R na množině atributů C, kde $C \subseteq A$
 - $R[C] = \text{def}\{u[C]|u \in R\}$ (výběr atributů)
- přirozené spojení = relací R(A) a S(B) s výsledkem T(C)
 - $R * S = T(A \cup B)$ (výběr n-tic = rovnosti na všech průnikových attributech A a B)
- obecné spojení

- $R[t_1 \Theta t_2]S(\Theta - podminka =, >, < \dots)$ (výsledek má všechny atributy včetně duplikací)
- přejmenování atributu
 - $t \rightarrow alias$
- množinové operace
 - sjednocení
 - průnik
 - rozdíl
 - kartézský součin

Příklad: *Hvězdy (herci), které hrají ve filmech promítaných v kině Mír: (nejsou zavorky!)*
`MA_NA_PROGRAMU(NazevK = 'Mir')[JmenoF,Datum]*FILM[Herec→Hvezda]`

Antijoin = podmnožina n-tic z R, které nejsou spojitelné s žádnou n-ticí z S. Minimální množinu operací tvoří: $\times$, selekce, projekce, $\rightarrow$, $\cup$, $\setminus$.

4 Polospojení

Interpretace $R <^* S$ = podmnožina n-tic z R, které jsou spojitelné s nějakou n-ticí z S. Polospojení není to samé jako Left/right join. Je to ověření podmínky, že mohou být spojeni. Relační dělení: Interpretace

Předpokládejme $R(A)$, $S(B)$

- **levé Θ -polospojení**

Značení: $R <_{t_1 \Theta t_2} S$

Definice: $R <_{t_1 \Theta t_2} S =_{def} (R[t_1 \Theta t_2]S[A])$

- **pravé Θ -polospojení**

Značení: $R[t_1 \Theta t_2 > S$

Definice: $R[t_1 \Theta t_2 > S =_{def} (R[t_1 \Theta t_2]S[B])$

- **levé přirozené polospojení**

Značení: $R <^* S$

Definice: $R <^* S =_{def} (R * S)[A]$

- **pravé přirozené polospojení**

Značení: $R * > S$

Definice: $R * > S =_{def} (R * S)[B]$

Lze chápat jako "syntaktickou zkratku" – spojení následované projekcí na A resp. B
 Skutečná implementace může být mnohem efektivnější.

$R \div S$ = výsledkem jsou všechny hodnoty x z R, které v R tvoří dvojici s každým prvkem y z S. Pomocí

prvků y z S se snažíme diskvalifikovat prvky x z R . Prvek x je diskvalifikován, pokud v R neexistuje ve dvojici s každým y z S . Výsledkem $R \div S$ jsou prvky x z R , které se diskvalifikovat nepodařilo.

- $R(x, y)$ a $S(y)$
- značení = $R \div S$
- definice = $R \div S =_{def} R[x] \setminus \{ \{ R[x] \times S \} \setminus R \}[x]$

D8''': {NazevK | kino něco hraje} \ {NazevK | kino nehraje některý film s Brando}
MA_NA_PROGRAMU[NazevK, JmenoF] ÷ {FILM(Herec = 'Brando')[JmenoF]}

5 SQL

- SQL = (Structured query language)
- slouží ke komunikaci s databázovým strojem
- říkáme, co chceme získat, ne jak se to má dělat
- intuitivně srozumitelný zápis
- připomíná jednoduché anglické věty

DDL Definiční jazyk – např. manipulace s tabulkama, integritní omezení **CREATE TABLE**

DML Manipulační jazyk – např. **SELECT**, **INSERT**, **UPDATE** apod.

DCL Jazyk na přístupy – **GRANT** <prikaz> ON <table> TO <user>

TCL Jazyk pro řízení transakcí – **COMMIT**, **ROLLBACK**

SELECT sloupce
FROM tabulky
 [WHERE podmínky]
 [ORDER BY řazení]

6 Integritní omezení

Omezení domény (tabulek)

- NOT NULL
- DEFAULT
- UNIQUE
- PRIMARY KEY
- REFERENCES
- CHECK

Okamžik kontroly IO, dočasné vypnutí/zapnutí IO:

- možnosti stanovit při deklaraci integritního omezení čas, kdy se má kontrolovat

- kontrolu IO lze definovat jako odložitelnou až na konec transakce
- v rámci session pak lze stanovit, zda IO kontroluje IMMEDIATE nebo až na konci transakce
- Oracle dovoluje v příkazu ALTER TABLE také IO dočasně vypnout/zneplatnit DISABLE/ENABLE CONSTRAINT
- zpětně zapnutí IO pak může/nemusí vyžadovat kontrolu platnosti dat již vložených v databázi