

## Databáze jednoduše pro prváky

### Základní pojmy

- atribut – sloupec tabulky
- doména – množina hodnot atributu
- uspořádaná ntice – řádek tabulky
- relační schéma – neprázdná uspořádaná ntice, záhlaví tabulky
- relace – množina uspořádaných ntic, jedna tabulka

### Klíče

- super klíč – množina atributů, které jednoznačně definují entitu
- kandidátní klíč – minimální super klíč
- primární klíč – vybraný kandidátní klíč
- cizí klíč – atribut, který musí existovat v jiném klíči

### Normální formy

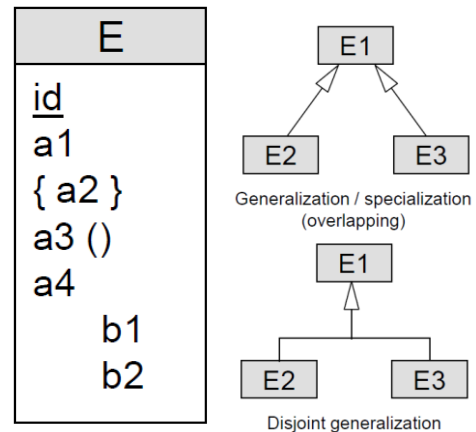
- první normální forma (1NF) – atributy jsou atomické
- druhá normální forma (2NF) – atributy jsou závislé na celém kandidátním klíči
- třetí normální forma (3NF) – všechny neklíčové atributy vzájemně nezávislé
- Boyceho–Coddova n. forma (BCNF) – levostranné funkční závislosti jsou kandidátním klíčem
- bezeztrátovost rozkladu – průnikem relací je jedna z relací,  $r = \Pi_{R1}(r) \bowtie \Pi_{R2}(r)$
- zachování funkčních závislostí – uzávěr závislostí je stejný,  $(F_1 \cup F_2 \cup \dots \cup F_n)^+ = F^+$

### Relační algebra

- 6 operací – selekce, projekce, přejmenování, sjednocení, rozdíl, kartézský součin, (přiřazení)
- bonus – zobecněná projekce, přirozené vnitřní spojení, levé a pravé vnější spojení a agregace
- přirozené vnitřní spojení (příklad) –  $r \bowtie s = \Pi_{r.A, r.B, r.C, r.D, s.E} (\sigma_{r.B=s.B \wedge r.D=s.D} (r \times s))$
- levé vnější spojení –  $r \Join s = (r \bowtie s) \cup ((r - \Pi_R(r \bowtie s)) \times \{(null, \dots, null)\})$
- pravé vnější spojení –  $r \Join s = (r \bowtie s) \cup (\{(null, \dots, null)\} \times (s - \Pi_S(r \bowtie s)))$
- průnik pomocí rozdílu –  $r \cap s = r \setminus (r \setminus s)$

## E-R diagram

- obdélník – tabulka – entitní množina
- kosočtverec – vztah
- čára bez šipek – násobnost M:N, obě šipky 1:1
- dvojitá šipka a dvojitý kosočtverec – slabá entita
- disjunktční / překrývající se množiny
  - může entita patřit do více entitních množin?
- úplná / částečná specializace množiny
  - musí entita patřit do jedné z nižších množin?



## Armstrongovy axiomy:

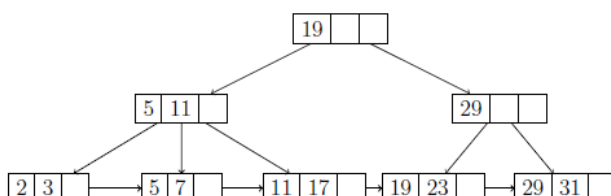
- testování funkčních závislostí při rozkladu
- jestli  $\beta \subseteq \alpha$ , pak  $\alpha \rightarrow \beta$  (reflexivita)
- jestli  $\alpha \rightarrow \beta$ , pak  $\gamma\alpha \rightarrow \gamma\beta$  (rozšíření)
- jestli  $\alpha \rightarrow \beta$  a  $\beta \rightarrow \gamma$ , pak  $\alpha \rightarrow \gamma$  (tranzitivita)

## Ukládání dat

- RAID: 0 žádná duplikace, 1 duplikace všeho, 2 paritní bit, 3 bitově prokládaná parita, 4 blokově prokládaná parita, 5 paritní blok na modulo disku, 6 dva kontrolní součty (drahé)
- řídký index – index uložen v souboru jen pro některé klíče
- hustý index – index uložen pro všechny klíče

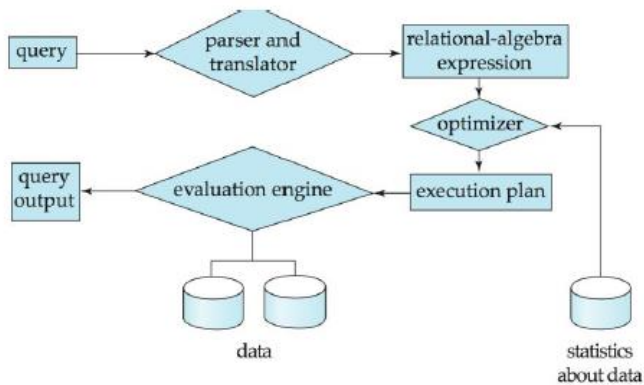
B<sup>+</sup>- strom

- větvení 4 ( $n = 4$ ) – maximální počet ukazatelů 4, hodnot 3
- všechny cesty od kořene k listům mají stejnou délku
- ukazatel před hodnotou ukazuje na prvky menší jak hodnota
- poslední ukazatel ukazuje na zbytek stromu
- pravé hodnoty uloženy až v listech, na rozdíl od B-stromu
- vnitřní uzly mají alespoň polovinu ukazatelů, listy polovinu hodnot



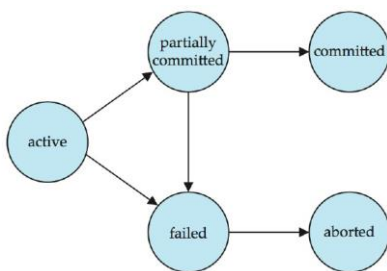
## Joiny a vyhodnocení

- nested join – projdi každý s každým a zkontroluj podmínku, univerzální ale pomalé
- hash join – vytvoř z obou primárních klíčů hashe a kyblíky, porovnávej kyblíky a hashe, lepší než nested join, nelze provést vždy
- merge join – seřaď obě relace na základě jejich primárního klíče, pospoj je (merge)
- optimalizace relační algebry – pravidlo ekvivalence – prohodit operace, sloučit je s ANDem, ...
- vyhodnocení dotazů – parsování, převod do algebry, optimalizace se statistikami, vytvoření plánu, vyhodnocení dat, výsledek



## Rozvrhy

- stavy transakce – aktivní (poč. stav), částečně provedená, potvrzená nebo chybující, zrušená
- rozvrh (plán) – posloupnost instrukcí ve kterém jsou transakce vykonány
- sériový rozvrh – provádění transakcí jedné po druhé, databáze zůstává v konzistentním stavu
- serializovatelnost současně běžících transakcí – provedení transakcí bez porušení konzistence
- předpokládáme, že v rozvrhu instrukce provádí výpočty na datech mezi čtením a zápisem
- konfliktní instrukce – pokud zápis (write) hodnoty **n**enásleduje hned po jejím čtení (read)
- konfliktně ekvivalentní rozvrhy – pokud jeden rozvrh lze převést na druhý (viz obrázek)
- konfliktně serializovatelný rozvrh – pokud rozvrh lze převést na sériový rozvrh



$T_1$	$T_2$	$T_1$	$T_2$
read (A)		read (A)	
write (A)		write (A)	
	read (A)	read (B)	
	write (A)	write (B)	
read (B)			read (A)
write (B)			write (A)
	read (B)		read (B)
	write (B)		write (B)

### Fragmentace

- časté používání disku
- volné místo rozptýlené po celém disku
- možná defragmentace

### Materializace

- drahý způsob optimalizace
- spočítej mezivýsledek, ulož a opakuj
- pipelines – předávej částečné výsledky hned na další zpracování, ne vždy použitelné

### SQL příkazy

- datové typy (data type) – VARCHAR(n), INT, FLOAT(n), DATETIME, BLOB, ...
- integritní omezení (constraint) – NOT NULL, UNIQUE, PRIMARY\_KEY, FOREIGN KEY, CHECK, DEFAULT, INDEX, ...
- pohled (view) – virtuální relace, může některé atributy skrýt, zlepší přehlednost a použití
- spouštěč (trigger) – automatický příkaz, vedlejší efekt, má podmínku spuštění a vlastní akci
- DELETE TABLE smaže řádky tabulky, DROP TABLE smaže tabulku i se schématem
- **SELECT FROM INNER/FULL JOIN ON/USING WHERE GROUP BY HAVING ORDER BY DSC**
- **INSERT INTO** zamestnanci (jmeno, mesto, stat) **VALUES** ('Pepa', 'Brno', 'Česko')
- **UPDATE** zamestnanci **SET** jmeno = 'Josef', City= 'Praha' **WHERE** ID = 1
- **DELETE FROM** zamestnanci **WHERE** jmeno = 'Josef'
- **CREATE VIEW** zamestnanci **SELECT** jmeno, prijmeni **FROM** zamestnanci\_vse
- **CREATE TABLE** tabulka (id INT PRIMARY\_KEY, jmeno VARCHAR(20), ...)
- **CREATE INDEX** muj\_index **ON** zamestnanci(jmeno, mesto, ...)
- **CREATE ASSERTION** tvrzeni **CHECK** podmínka
- **CREATE TRIGGER** inc **BEFORE INSERT ON** ucet **FOR EACH ROW SET** @s = @s + **NEW**.amount

### Navazující studium:

- Teorie od Dohnala: <https://www.fi.muni.cz/~xdohnal/lectures/PB154/>
- SQL online: <http://mufin.fi.muni.cz/projects/PB154/index.php>  
<http://mufin.fi.muni.cz/projects/PB154teach/index.php>