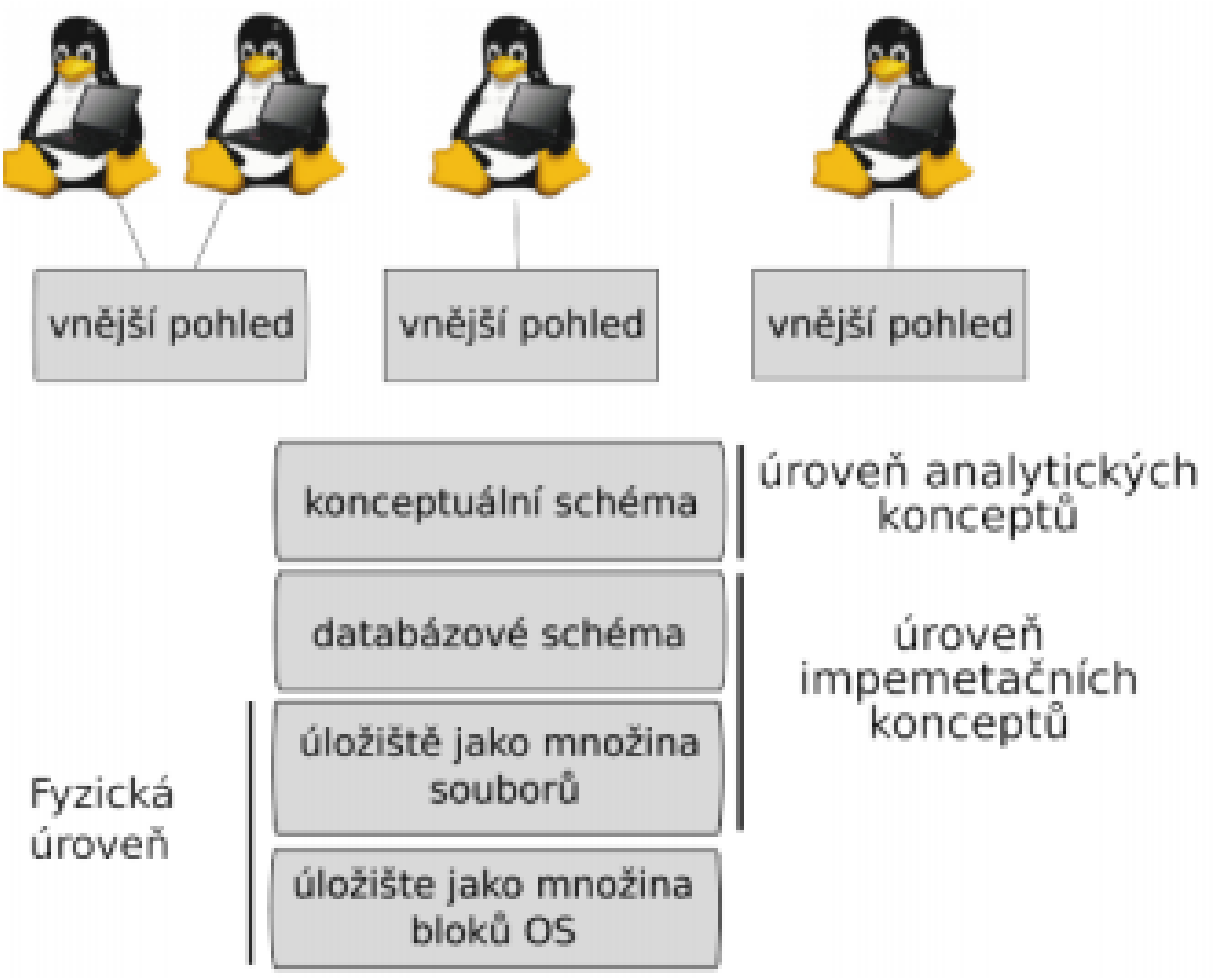
DBS – BI-SPOL-11

3 úrovně pohledu na data (konceptuální, implementační, fyzická). Struktury pro ukládání dat v relačních databázích s ohledem na rychlý přístup k nim (speciální způsoby uložení, indexy apod.)

Obsah

1 3 úrovně pohledu na data



Konceptuální Modelování reality (Obvykle se zachycuje se UML diagramem nebo ER modelem). Snaží se nebýt ovlivněna prostředky řešení.

Implementační Konkrétní databázový model, konstrukční dotazovací a manipulační prostředky (relační, objektová, síťová, hierarchická, XML, …)

Fyzická Sekvenční soubory, indexy, clustery apod.

2 Konceptuální modelování databází

- společné chápání objektu aplikace uživateli a projektanty
- integrace několika uživatelských pohledů
- výsledek je vstupem do realizace DB
- slouží jako dokumentace

3 Impplementační

- nejnižší míra abstrakce
- v této fázi probíhá realizace datové struktury, popsané v konceptuálním modelu
- model je zde transformován do modelu odpovídající konkrétní technologii
- musí zohledňovat všechny dostupné prostředky a možnosti
- popisuje, čím je datový obsah systému, popsaný konceptuálním a strukturálním modelem, realizován

4 Fyzický pohled

4.1 Struktury pro ukládání dat v relačních DB s ohledem na rychlý přístup k nim (speciální způsoby uložení, indexy apod.)

4.1.1 Heap

- nové záznamy přidány do libovolného prázdného místa
- žádné uspořádání
- hledání je O(n)

4.1.2 Heap s indexy

- záznamy jsou uspořádány
- víme, když už můžeme ukončit hledání

4.1.3 Cluster index

- index pages
- struktura už obsahuje samotné záznamy
- můžeme mít jenom jeden clustered index nad stejnými daty

4.1.4 Noncluster index

- ukazatele do samotných záznamů
- libovolná organizace indexu (ROW ID)

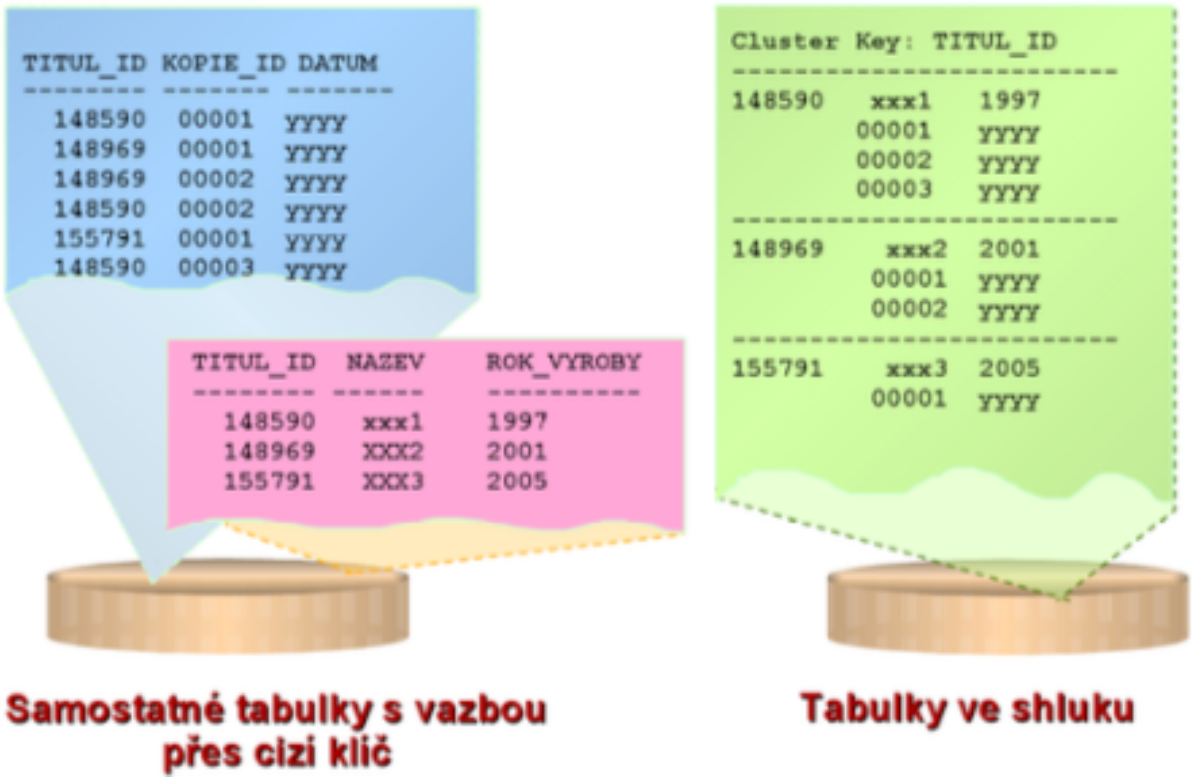
4.1.5 Bitmapové indexy

- binární matice
- předem vypočítané odpovědi na jednoduché otázky (true/false), a to pro každý záznam
- DLM operace velmi drahé
- spíš pro DSS (ne OLTP)
- vhodné pro záznamy s velmi neunikátními položkami

DSS = decission support system - velká rozhodnutí, založena na historických datech
OLTP = online transaction processing - aktuální data, každodenní transakce

4.1.6 Shluk

Tabulky dané do jednoho shluku.



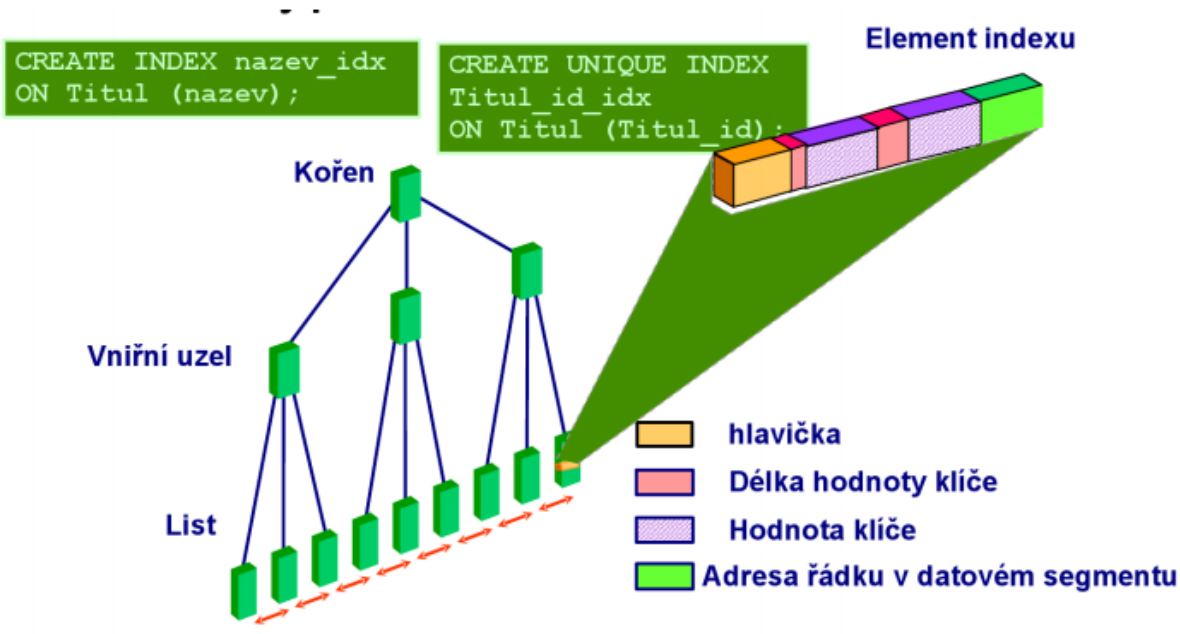
4.1.7 Index typu B*-Tree

- kořen má nejméně 2 potomky, pokud není listem
- každý uzel kromě kožene a listu má nejméně [m/2] a nejvýše m potomků
- každý uzel má nejméně [m/2] - 1 a nejvíce m - 1 datových záznamů
- všechny cesty ve stromě jsou stejně dlouhé
- data v nelistovém uzlu jsou organizována
- listy obsahují úplnou množinu klíču a mohou se lišit strukturou

5 Důležité poznatky

- (relační) databáze bez indexu nefungují rozumně - indexy jsou nutné (pro větší data)
- DB stroj často některé indexy vytváří automaticky kvůli kontrole IO (integritní omezení)
- v OLTP se nejčastěji používají indexy na bázi B-stromů (tam kde jsou data unikátní)
- kde jsou data velmi neunikátní a potřebují se indexovat, tam se používají bitmapové indexy
- indexy je třeba udržovat (zjednodušen+ - ušetřím na dotazech, platím více při DML)
- klič indexu (indexované atributy) může být složený

```
CREATE INDEX nazev_idx on Titul (nazev);
CREATE UINIQUE INDEX titul_id_idx on Titul (titul_id);
```



- index může být unikátní/neunikátní