

Objektově-relační databáze

Z FITwiki

Databáze

Doporučuju přečíst [1]

Obsah

- 1 Databáze
 - 1.1 Relační databáze
 - 1.2 Objektové databáze
 - 1.3 Objektově-relační databáze
- 2 Standardy O-R databází
 - 2.1 Standard SQL-1999
 - 2.1.1 Nové relační rysy
 - 2.1.2 Objektové rysy
 - 2.2 Standard SQL-2003

(http://www.fit.vutbr.cz/study/courses/DSI/public/pdf/nove/12_2.pdf)

Relační databáze

Od 80.let získaly převahu nad síťovými a hierarchickými.

Vlastnosti:

- **standardizovány** - SQL (od roku 1986)
- relační model dat, formálně zavedeno od Codda
- představovány tabulkami obsahujícími relace
- podpora výrobců DB produktů

Model dat:

- relační
- kolekce tabulek
- vztahy tabulek vyjádřeny pomocí cizích a kandidátních klíčů
- navigace po tabulkách pomocí kurzoru

Nevýhody:

- velmi omezená množina datových typů hodnot
- pro vztahy M:M je nutná vazební tabulka
- žádné reference či ukazatele

Dotazovací jazyk:

- SQL
- neprocedurální, deklarativní

Objektové databáze

Umožňují modelování a vytváření perzistentních dat jako objektů. Čistě objektové systémy nebyly standardizovány a proto se nerozšířily.

Na konci 80.let vliv paradigmatu OO programování -> vznik OO DB (tj. perzistentního uložení objektů).

Vlastnosti:

- modelování a vytváření perzistentních dat jako objektů
- nenahrazují, nýbrž doplňují relační (kombinace obou se nazývá objektově-relační DB)

Výhody:

- vztahy M:M lze vytvářet přímo
- navigace po objektové struktuře pomocí referencí prostřednictvím OID
- atributy objektů mohou být jiné objekty → složité typy (ADT - abstraktní datové typy)

Model dat:

- objektový
- **neexistuje všeobecně přijatý standard obdobný SQL** (ODMG-93 byl pokusem)
- jednoznačný OID (objektový identifikátor) pro každý perzistentní objekt
- podpora abstraktních datových typů (ADT), zapouzdření a polymorfismu
- atributy objektů mohou být jiné objekty → složité typy
- vztahy objektů pomocí referencí

Dotazovací jazyk:

- většinou běžné objektové jazyky
- snaha o standardizaci (jazyk OQL od ODMG)

V objektové databázi by SŘBD neměl povolit: pokud výsledek jedné z operací je objekt s parametry, který nelze v daném SŘBD uložit (výsledek operace nelze popsat objekty v DB).

Objektově-relační databáze

Cílem je spojit výhody relačního a objektového modelu.

Model dat:

- Tabulky **nemusejí** být normalizované (porušují první normální formu)
- obecnější (vnořené) relace (nested relational model)
- data stále v tabulkách, ale hodnoty mohou mít bohatší strukturu (ADT)
- ADT zapouzdřuje data i operace
- zavedeno OID umožňující definovat nové typy vztahů mezi tabulkami
- navigace pomocí kurzoru i pomocí referencí

Dotazovací jazyk:

- ve standardu SQL-1999 (viz dále)

Výhody:

- snaha o obohacení tabulek o objektovou orientaci
- navigace po tabulkách pomocí kurzoru i pomocí referencí

Standardy O-R databází

Standard SQL-1999

Nové relační rysy

- nové datové typy (LOB (Large Objects) – BLOB (Binary LOB), CLOB (Character LOB))
- nové predikáty (SIMILAR – obdoba LIKE, ale s reg. výrazy)

WHERE verze SIMILAR 'SQL-(86|89|92|1999|2003)|SQL(1|2|3)'

- rekurzivní dotaz
- SAVEPOINT – pro částečný návrat transakce
- trigger
- ROW – možnost definování složeného sloupce

```
CREATE TABLE Osoba (jmeno ROW(krestni VARCHAR(20), prijmeni VARCHAR(30)), ...)
SELECT O.jmeno.krestni FROM Osoba O
```

Objektové rysy

- **uživatелеm definované typy (UDT)**
 - obdoba třídy
 - atributy, metody
 - jednoduchá dědičnost

```
CREATE TYPE zamestnanec_t
UNDER t_osoba
AS (os_cislo INTEGER,
    plat REAL )
INSTANTIABLE
NOT FINAL
REF (os_cislo)
INSTANCE METHOD
Zvys_plat(abs_proc BOOLEAN, castka REAL)
RETURNS REAL
```

- typ sloupce tabulky může být UDT
 - v tomto případě nemají OID

```
CREATE TABLE Zamestnanci_fakulty (
id INTEGER PRIMARY KEY,
zam zamestnanec_t
```

- odkaz na atributy přes funkce nebo tečkovou notací

```
WHERE zam.plat > 20000
```

- **typové tabulky**
 - tabulka s řádky nesoucími hodnoty strukturovaného UDT
 - každý řádek má OID (hodnota typu REF)
- **typ REF**
 - odkazy na řádky v typové tabulce
 - mají vždy rozsah (scope)

```
vedouci REF(zamestnanec_t)
```

- umožňují zpřístupnit atributy hodnoty strukturovaného typu, kterou identifikují

```
SELECT vedouci -> jmeno
```

Standard SQL-2003

- Nově podpora multimnožin: typ MultiSet je odvozen od typu Collection
- Podpora pokročilých Multimnožin: pomocí operátoru Porovnávání, SET, UNION, INTERSECTION a pod.

- Definuje 2 nové typy kolekcí: Array a Multiset
- Změna u UDT: Může být FINAL nebo NOT FINAL pro zákaz dědění resp. možnost dědění

Citováno z „http://wiki.fituska.eu/index.php?title=Objektov%C4%9B-rela%C4%8Dn%C3%AD_datab%C3%A1ze&oldid=13221“

Kategorie: Pokročilé databázové systémy | Státnice MGM | Státnice 2011

- Stránka byla naposledy editována 28. 5. 2016 v 11:44.