

# TESTOVÁNÍ VLIVU NÁVRHU SCHÉMATU DATABÁZE NA EFEKTIVITU VYHODNOCOVÁNÍ DOTAZŮ

**Ondřej Daněk**

*Ostravská univerzita, Přírodovědecká fakulta, Katedra informatiky a počítačů, 30. dubna  
22, 701 03 Ostrava, [ondrej.danek@outlook.cz](mailto:ondrej.danek@outlook.cz)*

## Abstrakt

Příspěvek se zabývá porovnáním rychlosti provádění dotazů při využívání objektově relačních prvků v databázi. Úvod je zaměřen na popis objektově relačních vlastností a jejich implementaci v některých databázových systémech. Další popisuje návrh modelů, dotazů a realizaci měření rychlosti vyhodnocování těchto dotazů. Poslední část se věnuje vyhodnocení výsledků a jejich porovnání na vybraných databázových systémech.

**Klíčová slova:** relační databáze; objektově relační databáze; testování; vyhodnocování dotazů;

## Úvod

Databázové systémy nás obklopují ze všech stran už několik desítek let. Zatím nejrozšířenějším modelem pro ukládání dat byl již od počátků databází relační datový model, který navrhl roku 1970 E. F. Codd. Od devadesátých let se s nástupem objektově orientovaného programování objevují první objektové databáze a snahy o jejich zavedení v komerčním světě. Ačkoliv mají objektové databázové systémy řadu výhod, mezi které patří zejména rozsáhlá podpora objektově orientovaného programování, mají rovněž řadu nevýhod. Řešením se zdá být zavedení objektově relačních databázových systémů, které se snaží spojit výhody obou zmíněných modelů.

Za cíl práce jsem zvolil analýzu a vyhodnocení toho, zda se při použití objektově relačních vlastností v databázi sníží její efektivita vyhodnocování dotazů. Jako prostředek k dosažení tohoto cíle jsem navrhl experiment, který jsem provedl na třech konkrétních databázových systémech.

## Implementace objektově relačních vlastností v systémech

Objektově relační databáze bývají řazeny do třetí generace databázových systémů. Prvotní požadavky na tyto databáze jsou definovány v *Third Generation Database System Manifesto* [1], na který později navázal Michael Stonebraker roku 1996 [2]. Objektově relační databáze jsou zde chápány jako rozšíření relační koncepce o některé objektové rysy. Důvodů pro zachování relační koncepce je více. Zejména šlo o zachování zpětné kompatibility s relačními databázemi a s tím spojené bezproblémové zapojení ORD do provozu. Dalšími důvody bylo například zachování jazyka SQL a to, že společnosti nechtěly zahodit obrovské investice do vývoje relačních databází.

Dle knihy Michaela Stonebrakera můžeme považovat za dobrý objektově relační systém ten, který obsahuje následující vlastnosti.

První vlastností je rozšíření základních datových typů databáze o další základní datové typy (zde nemluvím o strukturovaných datových typech). Toto rozšíření základních datových typů je v systémech řešeno buď definováním nových typů, nebo použitím modulů, které základní datové typy obsahují. Například u databázi společnosti Oracle se tyto moduly nazývají cartridge.

Druhá a zřejmě ta nejdůležitější vlastnost je použití strukturovaných datových typů. Strukturované datové typy jsou jakousi analogií ke třídám z objektově orientovaného programování. Byly definovány ve standardu SQL 1999 (někdy též SQL 3).

Třetí vlastností je dědičnost mezi strukturovanými datovými typy. Díky této vlastnosti může datový typ (subtyp) dědit od jiného datového typu (supertypu) jeho vlastnosti. Databázové systémy podporují většinou jednoduchou dědičnost, přičemž dovoluují subtypu dědit jak atributy, tak metody.

Poslední vlastností jsou pravidla. Pravidla pomáhají chránit integritu dat a usnadňují údržbu databázového systému. Relační databáze sice podporují jistou formu pravidel také (triggery), ale pravidla v objektově relačních databázích mají více možností, které jsou při použití objektového přístupu zapotřebí.

Výsledek analýzy pěti nejpoužívanějších relačních databázových systémů je zobrazen v tabulce 1, kde vidíme, které systémy implementují objektově relační vlastnosti pomocí jazyka SQL. Z výsledků je patrné, že za objektově relační můžeme považovat Oracle Database, PostgreSQL a IBM DB2.

**Tabulka 1.** Implementace objektově relačních vlastností v reálných systémech

	Rozšíření základních datových typů	Strukturované datové typy	Typová dědičnost	Pravidla
Oracle Database	ANO	ANO	ANO	NE
MySQL	NE	NE	NE	NE
MS SQL Server	NE	NE	NE	NE
PostgreSQL	ANO	ANO	NE	ANO
IBM DB2	ANO	ANO	ANO	ANO

### Návrh modelů a dotazů

Aby bylo možné vhodně porovnávat rychlost vyhodnocení dotazů, bylo potřeba navrhnout relační a objektově relační modely pro každý systém. Všechny modely vychází z jednoho konceptuálního modelu a konkrétní relační a objektově relační modely byly tedy navrženy nad stejnou doménou. Při návrhu objektově relačních modelů jsem se snažil co nejvíce využít vlastnosti popsané v předchozí části textu. Jelikož se implementace objektově relačních vlastností v systémech liší, tak se v některých částech liší i objektově relační modely.

Důležitým kritériem je, aby dotazy pokrývaly co nejširší spektrum možností databáze. Snažil jsem se tedy využít různé druhy propojení tabulek, agregační funkce, vnořené i jednoduché selekty, různé podmínky, klauzule GROUP BY, HAVING, EXISTS, ANY atd. a u objektově relační databáze také typové metody, kolekce a pole. Celkem jsem navrhl deset dotazů pro každý model, které vrací na všech modelech shodné výsledky.

### Měření

Pro testování rychlosti odpovědi na dotazy jsem vytvořil aplikaci ve vývojovém prostředí Visual Studio 2013. Součástí programu je uživatelské rozhraní, které staví na technologii WPF (Windows Presentation Foundation). Tato technologie používá značkovací jazyk XAML, který umožňuje přehledné oddělení vzhledu aplikace od aplikační logiky, jež je v tomto případě psána v jazyce C# modely.

Aplikace umožňuje zvolit napevno vložený dotaz, který provede 100 krát na relační databázi a následně také na databázi objektově relační. Mezi jednotlivými dotazy aplikace nevyprazdňuje cache paměť, jelikož tento způsob vrací reálnější výsledky. Více o testování dotazů a s tím související využití cache paměti píše databázový specialista společnosti Oracle Tom Kyte [3].

Výsledky zapisuje aplikace do souboru ve formátu CSV, aby byl bezproblémový import těchto výsledků do statistického programu NCSS.

### Vyhodnocení výsledků

Výsledky měření jsem vyhodnotil pomocí statistického programu NCSS. Testoval jsem pro každý dotaz a systém hypotézy o shodě středních hodnot pomocí párového t-testu.

V tabulce 2 jsou zaznamenány výsledky testování hypotéz na všech vybraných dotazech a systémech. Hypotéza o shodě středních hodnot je označena jako  $H_0$ . Hypotéza  $H_R$  říká, že střední hodnota odpovědí a dotaz je menší u relačního modelu než u objektově relačního, opakem je hypotéza  $H_{OR}$ . Dotazy jsou označeny D1 až D10. Heslem ANO jsou označeny hypotézy na dotazech, které vyhodnotil program NCSS jako platné (hypotézu přijímáme), heslo NE značí opak. Z výsledků je patrné, že mezi jednotlivými systémy a dotazy jsou mnohdy významné rozdíly.

**Tabulka 2.** Shrnutí výsledků párových t-testů

	Oracle			DB2			PostgreSQL		
	$H_0$	$H_R$	$H_{OR}$	$H_0$	$H_R$	$H_{OR}$	$H_0$	$H_R$	$H_{OR}$
<b>D1</b>	NE	<b>ANO</b>	NE	NE	<b>ANO</b>	NE	NE	NE	<b>ANO</b>
<b>D2</b>	NE	<b>ANO</b>	NE	NE	NE	<b>ANO</b>	NE	NE	<b>ANO</b>
<b>D3</b>	NE	<b>ANO</b>	NE	NE	NE	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>
<b>D4</b>	NE	NE	<b>ANO</b>	NE	NE	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>
<b>D5</b>	NE	<b>ANO</b>	NE	NE	NE	<b>ANO</b>	NE	NE	<b>ANO</b>
<b>D6</b>	NE	<b>ANO</b>	NE	NE	NE	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>
<b>D7</b>	NE	<b>ANO</b>	NE	NE	NE	<b>ANO</b>	NE	<b>ANO</b>	NE
<b>D8</b>	NE	<b>ANO</b>	NE	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	NE	NE	<b>ANO</b>
<b>D9</b>	NE	<b>ANO</b>	NE	NE	<b>ANO</b>	NE	NE	<b>ANO</b>	NE
<b>D10</b>	NE	<b>ANO</b>	NE	NE	NE	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>	<b>ANO</b>

### Závěr

Statisticky vyhodnocené výsledky nedávají jasnou odpověď na otázku, zda při použití objektově relačních prvků klesá výkonnost databáze pro vyhodnocování dotazů. Skutečnost, že výsledky nebyly jednostranné, však vybízí k dalšímu zkoumání. Rozhodně jsem tedy nedospěl k názoru, že relační databáze je výkonnější, ale ani naopak. Na základě vypracování této práce si dovoluji tvrdit, že i když nebyly výsledky zcela jednoznačné, tak jsou objektově relační databáze perspektivní díky širším možnostem a uživatelské přívětivosti. Práce s nimi však v komplikovanějších úkolech vyžaduje precizní znalosti pro dosažení kýžených výsledků.

### Literatura

- [1] STONEBRAKER, M., A DALŠÍ. *Third Generation Database System Manifesto*. New York, NY, USA : ACM SIGMOD Record, 1990.
- [2] STONEBRAKER, MICHAEL, BROWN, PAUL A MOORE, DOROTHY. *Objektově-relační SRBD: analýza příští velké vlny*. [překl.] Jiří Felbáb. 1. české vydání. Praha : BEN, 2000. str. 279. 80-86056-94-5.

[3] KYTE, TOM. On Procedures, Flushes, and Writes. *Oracle Corporation*. [Online] Oracle Corporation. [Citace: 30. Duben 2015.] <http://www.oracle.com/technetwork/issue-archive/o43asktom-094944.html>.

### **Abstract**

This contribution deals with comparison of query execution speed in database which uses object-relational attributes. The first part focuses on description of object-relational attributes and their implementation in some database systems. Next part describes model design, query design and realization of query response measurement. The last part is dedicated to evaluation of results and their comparison on selected database systems.