

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačních technologií**



**Teze diplomové práce**

**Optimalizace výkonu databázových systémů**

**Jan Mikolášek**

© 2017 ČZU v Praze

## **Souhrn**

Hlavním cílem této práce je navrhnout vhodné prostředky pro ladění výkonu relačních databází. Relační databáze je základem takřka každého informačního systému, a její výkon ovlivňuje celkovou výkonnost tohoto systému. Vlastní výkonnost relační databáze je dána už při samotném návrhu fyzické i logické architektury databáze a následně také pravidelným ověřováním správnosti návrhu a reakcí na měnící se požadavky. Cílem této práce je stanovit správný postup při návrhu relační databáze, určit klíčové oblasti, na které je potřeba se zaměřit při návrhu a průběžném monitoringu databázového systému. Tyto poznatky budou následně prakticky ověřeny na fiktivní databázi s využitím databázových systémů od společností IBM, Oracle a Microsoft. V práci je nejdříve stanovena metodika, podle které bude postupováno k dosažení hlavního cíle této práce. V následující části jsou popsány teoretické základy jednotlivých komponent relační databáze a také vybraných částí, které mají vliv na výkon relační databáze. V praktické části bude vytvořena fiktivní databáze a bude zkoumáno její chování v různých situacích. V závěru práce budou využity výsledky praktické části ke stanovení správného procesu návrhu relační databáze poskytující optimální výkon při různých způsobech implementace.

**Klíčová slova:** výkonnost, relace, databáze, tabulka, databázový index, hledání, třídění, komprese, optimální cesta, dotaz, časová složitost

## **Cíl práce**

Hlavním cílem této práce je popsat a ověřit postupy pro ladění výkonu relační databáze ve zvolených oblastech. K dosažení tohoto cíle vedou následující dílčí kroky, jakými jsou rozbor a popis typických oblastí ovlivňujících výsledný výkon relační databáze, popis nástrojů správce databáze pro analýzu výkonu, návrh relační databáze a jejich struktur s ohledem na požadavky pro dosažení optimálního výkonu a praktické ověření teoretických informací v praxi za použití nejrozšířenějších systémů pro správu relačních databází, kterými jsou Oracle, IBM DB2 a Microsoft SQL server.

## **Metodika**

Metodika tvorby této práce je v úvodu založena na studiu odborné literatury k získání informací o možnostech současných databázových systémů pro optimalizaci výkonu. Tyto nabyté znalosti budou následně použity a zkoumány v praktické části

V úvodu praktické části bude navržena fiktivní databáze, která slouží pro ověření získaných teoretických poznatků z oblasti ladění výkonu relačních databází. Tato databáze

bude navržena s ohledem na vybrané prostředky pro ladění výkonu, tak aby bylo možné tyto prostředky implementovat a ověřit jejich skutečné reálné využití v praxi. Takováto databáze bude postupně implementována do všech vybraných databázových systémů, na kterých budou prováděna měření. Pro potřeby měření bude vyvinuta speciální aplikace, která bude generovat reálnou zátěž pomocí zasílání specifických dotazů pro každý typ měření volitelným počtem vláken simulující reálnou práci uživatelů. Před vlastním měřením bude vygenerována identická sada dat pro jednotlivé databázové systémy.

V rámci vlastního měření bude pro každý implementovaný prostředek pro optimalizaci výkonu relační databáze provedena sada měření na všech vybraných databázových systémech. Pro objektivnější výsledky měření bude každé měření provedeno několikrát a následně vybrán medián naměřených hodnot jako výsledný čas zpracování dotazu. V závěru bude provedeno zhodnocení všech porovnávaných prostředků a dáno doporučení na využití těchto prostředků pro jednotlivé typy implementací relačních databází.

## **Praktická část**

Po vlastní implementaci všech databázových systémů včetně navržené databázové struktury bude zahájena fáze generování testovacích dat. Po naplnění vybraných databázových systému daty bude zahájeno měření jednotlivých oblastí pro optimalizaci výkonu relačních databází. V rámci této práce byly vybrány oblasti vyrovnávací paměti, statických versus dynamických dotazů, databázových indexů, tabulkový partitioning a komprese dat v tabulkách. V rámci testování byly primárně využívány dvě tabulky reprezentující vztah zákazník – faktura, kdy tabulky obsahovaly zhruba 14 respektive 148 milionů záznamů.

## **Vlastní měření**

### **Vyrovňovací paměť**

Toto měření bylo z pohledu poskytovaných možností jednotlivých databázových systémů nejsložitější. Každý z vybraných systémů umožňoval jinou možnost konfigurace velikosti vyrovnávací paměti, kdy nejvyšší možnosti ladění poskytoval IBM DB2, Microsoft SQL neposkytoval takřka žádné možnosti. V případě IBM DB2 je možné nastavit paměť až na úroveň konkrétního datového souboru, tedy vlastně až na úroveň konkrétní tabulky. Tento test byl prováděn pouze jednou, a to z toho důvodu, že efektu vyrovnávací paměti je dosaženo až po určitém kontinuálním běhu aplikace. Nejprokazatelnějších výsledků dosáhl databázový systém Oracle, kde bylo jasně patrné snižování času s narůstajícím počtem

dotazů. Průměrně byl snížen čas zpracování dotazu o 50ms, u databázového systému Microsoft SQL nebylo možné měření provést, neboť zde je možné definovat paměť pouze pro celou instanci databázového serveru. V případě následujících měření bylo naopak potřeba vliv vyrovnávací paměti eliminovat, a tudíž byl obsah paměti před každým měřením nutné vymazat.

### **Statické versus dynamické dotazy**

V rámci tohoto testu bylo prováděno pět měření na každém databázovém systému ve variantě statického dotazu a variantě dynamického dotazu. Výsledné průměrné časy zpracování dotazu na všech databázových systémech hovořily ve prospěch dynamické varianty dotazu, kde časy byly lepší v průměru o zhruba pět milisekund.

### **Databázový index**

Toto měření bylo prováděno v několika různých variantách vytvoření databázových indexů. Byl zkoumán rozdíl mezi složenými a jednoduchými indexy i pořadí jednotlivých sloupců tabulek v indexu. Jako vedlejší měření v rámci tohoto testu bylo provedeno porovnání rychlosti vkládání záznamů do tabulky s indexem a bez. Výsledkem tohoto měření bylo jednak to, že databázový index je vhodné v případě, kdy je to možné, vyrobit jako složený. Na pořadí sloupců nezáleží, což je častokrát zmiňováno jako důležitá součást při návrhu databázového indexu. Nejmarkantnější rozdíl mezi složeným indexem a jednoduchými indexy byl u databázového systému Oracle, kde se čas zpracování dotazu snížil zhruba 15x, u ostatních systémů nebyl tento rozdíl tak dramatický. V rámci vedlejšího měření bylo potvrzeno, že v případě neexistence jednoho databázového indexu je do tabulky možné vložit až dvakrát více záznamů za jednotku času.

### **Tabulkový partitioning**

V rámci tohoto testu byla prokázána vhodnost využití tabulkového partitioningu v situacích, kdy je možné rozdělit do logických celků podle určitého sloupce, který je použit i v rámci dotazů do databáze. Největší efekt byl zaznamenán u databázového systému IBM DB2, kdy bylo značné snížení průměrného času dotazu. Ostatní databázové systémy, takový rozdíl nevykazovaly, Microsoft SQL server dokonce nevykazoval žádné zlepšení.

### **Tabulková komprese**

V případě této oblasti byl zjištěn efekt zlepšení výkonu až od určitých, poměrně vysokých, kompresních poměrů. Patrného zlepšení výkonu bylo dosaženo při kompresním výkonu 10:1 a vyšším. Opět různé databázové systémy poskytovaly různé možnosti

nastavení komprese, od komprese řádku (IBM DB2) až po komprese celé stránky (Oracle, Microsoft SQL).

## **Závěr**

Z provedených měření je patrné, že každý z databázových systémů disponuje všemi popsányi prostředky pro optimalizaci výkonu relačních databází s různým efektem. Nicméně je možné říct, že až na kompresi dat, která se jeví jako spíše výhodná z pohledu úspory diskového prostoru je všechny ostatní prostředky možné použít pro ladění výkonu relační databáze. Je také zřejmé, že s přibývajícím počtem požadavků na databázi budou tyto rozdíly ještě znatelnější. V následující tabulce jsou uvedeny různé typy implementace relační databáze a doporučení implementace vybraných prostředků při ladění jejich výkonu.

	<b>Datový sklad</b>	<b>Provozní databáze s rozlišením aktivních a pasivních dat.</b>	<b>Provozní databáze bez rozlišením aktivních a pasivních dat.</b>
<b>Vyrovnávací paměť</b>	NE	ANO	NE
<b>Statický versus dynamický dotaz</b>	Statický	Dynamický	Dynamický
<b>Databázový index</b>	ANO (pokrytí většiny dotazů)	ANO (rozumný počet s ohledem na rychlost vkládání)	ANO (rozumný počet s ohledem na rychlost vkládání)
<b>Tabulkový partitioning</b>	NE	ANO	ANO
<b>Tabulková komprese</b>	ANO	Záleží na velikosti kompresního poměru	Záleží na velikosti kompresního poměru

## Seznam vybraných použitých zdrojů

- Bayer, R. (2002). *Organization and maintenance of large ordered indexes*. Springer Berlin Heidelberg.
- Codd, E. F. (1970). *A relational model of data for large shared data banks*. San Jose: IBM.
- Chaudhuri, S., & Narasayya, V. (2007). *Self-Tuning Database Systems: A Decade of Progress*. VLDB Endowment
- Lui, H. (2011). *Oracle database performance and scalability*. IEEE Computer Society Press. ISBN 9781118135532
- Niemann, T. (2008). *Sorting and searching algorithms*. Portland: epaperpress.com.
- Oracle. (9 2016). *Analyzing SQL with SQL tuning advisor*. Získáno 11. 3 2017, z Oracle help center: <https://docs.oracle.com/database/121/TGSQL/title.htm>
- Pokorný, J., & Valenta, M. (2013). *Databázové systémy*. Praha: České vysoké učení technické. ISBN 978-80-01-05212-9
- Powell, G. (2011). *Oracle performance tuning for 10gR2*. Elsevier Science. ISBN 9781555583453
- Rahayu, W., Leung, C., & Taniar, D. (2008). *High Performance Parallel Database Processing and Grid Databases*. John Wiley & Sons, Incorporated. ISBN 9780470107621
- Selinger, P., Astrahan, M., Chamberlin, D., Lorie, R., & Price, T. (1973). *Access Path Selection in a Relational Database Management System*. San Jose: IBM Research division.