

Vysoká škola ekonomická v Praze

Fakulta informatiky a statistiky

Katedra informačních technologií

Student : **Roman Lelek**

Vedoucí bakalářské práce : **Ing. et Ing. Tomáš Šalamon, MBA**

Recenzent bakalářské práce : **Ing. Petra Hradecká**

TÉMA BAKALÁŘSKÉ PRÁCE

Srovnání Firebird SQL se systémem Oracle Database

ROK: 2009

Prohlášení

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně a že jsem uvedl všechny použité prameny a literaturu, ze kterých jsem čerpal.

V Praze dne 7.12.2009

.....

podpis

Poděkování

Rád bych poděkoval panu Ing. et Ing. Tomáši Šalamonovi, MBA za hodnotné připomínky a čas, který mi v průběhu celé práce věnoval. Dále bych rád poděkoval své rodině za podporu při studiu.

Abstrakt

Tato bakalářská práce se zabývá porovnáním dvou databázových systémů. Jedná se o relační open source databázový systém Firebird a komerční verzi Oracle Express Edition.

V první části jsou oba databázové systémy teoreticky popsány podle několika hledisek, mezi něž patří cena a licence, funkcionalita a technologie, konzistence, řízení zálohy a obnovy dat, technická dokumentace a výkon. Čtenář tak obdrží informace využitelné při rozhodování mezi open source nebo komerčním řešením.

V druhé části je popsáno, jak byl v praktické rovině ověřen výkon obou databází v testovací aplikaci vytvořené pomocí jazyka Java. Firebird se ukázal být lepší variantou pro použití v jednouživatelských aplikacích (například aplikace pro mobilní telefony), víceuživatelských aplikací pro operační systém Windows a on-line sběru a analýze dat. Oracle je vhodné použít jako testovací prostředí vyšších verzí nebo také pro hlavní databázi malého projektu (například Wikipedie).

Přínosem této práce je to, že jsou zde shromážděny nejdůležitější informace o zástupcích dvou rozdílných databázových systémů. Jsou utříděny tak, aby mohly být v teoretické rovině porovnány z hlediska praktického využití, dále pak to, že jsou prakticky otestovány jejich výkony a zaznamenány výsledky o tom, jak rychle zpracovávají SQL příkazy. Informace tohoto druhu totiž nejsou dostupné ani v tištěných ani elektronických zdrojích.

Abstract

This bachelor thesis deals with comparing two different database systems. The first is a relational open source database system Firebird and the second is commercial solution Oracle Express Edition.

In the first part, both database systems are described from several perspectives – general characteristics, functionality and technology, consistency, data backup and recovery management, technical documentation, and performance. The reader is provided with information that he or she can use when deciding between an open source and a commercial solution.

In the second part of the thesis is described, how the both databases' performance is tested practically, in a test application created in Java.

Firebird has proven to be a better option for single-user applications (such as applications for mobile devices), multi-user applications for Windows and on-line data collection and analysis. Oracle should be used as a test environment of higher editions or as a master database for a small project (e.g. Wikipedia).

Contributions of this thesis is not only in gathering, analyzing and sorting information about two different database systems from various sources and comparing the theoretical level to terms of practical use, but mainly in testing of performance and providing the information how quickly handles specific SQL commands. This kind of information couldn't be found in any published or digital sources.

Obsah:

1. Úvod	1
2. Vymezení základních pojmů	4
3. Stanovení oblastí a kritérií porovnávání	6
3.1. Porovnáváné databázové systémy	6
3.1.1. Historie	6
3.2. Použitý hardware	7
3.3. Oblasti a kritéria porovnávání	8
3.3.1. Cena a licence	8
3.3.2. Funkcionalita a technologie.....	8
3.3.3. Konzistence	10
3.3.4. Řízení zálohy a obnovy dat	10
3.3.5. Technická podpora a dokumentace	10
3.3.6. Výkon.....	10
4. Porovnání na základě stanovených kritérií.....	11
4.1. Cena a licence	11
4.2. Funkcionalita a technologie	13
4.2.1. Omezení	13
4.2.2. Architektura.....	16
4.2.3. Instalace	27
4.2.4. Administrace	28
4.2.5. Datové typy	30
4.2.6. Databázové objekty.....	33
4.2.7. Zvláštní funkcionalita.....	38
4.3. Konzistence	41
4.4. Řízení zálohy a obnovy dat	46
4.5. Technická podpora a dokumentace	48

5.	<i>Porovnání výkonu</i>	51
5.1.	Příprava testování	51
5.2.	Postup testování	53
5.3.	Dosažené výsledky	54
5.3.1.	Výběry dat	55
5.3.2.	Operace s daty (přidávání, aktualizace, mazání)	56
5.3.3.	Rozšíření	57
6.	<i>Shrnutí dosažených výsledků</i>	58
7.	<i>Závěr</i>	64
8.	<i>Přílohy</i>	67
8.1.	Příloha 1 - Aktuální ceník licencí Oracle	67
8.2.	Příloha 2 – Použité SQL příkazy	68
8.2.1.	Výběry dat	68
8.2.2.	Operace s daty (aktualizace, vložení, smazání)	69
8.2.3.	Rozšíření	70
8.3.	Příloha 3 – Detailní rozbor výsledků	72
8.3.1.	Výběry dat	72
8.3.2.	Operace s daty (aktualizace, vložení, smazání)	74
8.3.3.	Rozšíření	76
9.	<i>Zdroje</i>	78
9.1.	Seznam použité literatury	78
9.2.	Seznam použitých obrázků	81
9.3.	Seznam tabulek	81

1. Úvod

Dnešní doba je charakteristická prudkým rozvojem informačních systémů, které vycházejí vstříc potřebám společností a dovedou uchovávat velké množství podnikových dat.

Dříve pro správu dat stačila pouze jednoduchá databáze pro jednoho uživatele, v současné době však podniky potřebují ukládat denně velké objemy dat. Tyto společnosti proto hledají takové databázové systémy, které by uspokojily jejich náročné požadavky. Nejde přitom jen o velké datové objemy, ale také o okamžitou dostupnost, víceuživatelský přístup a v neposlední řadě o dostatečné zabezpečení.

Podniky (uživatelé) mají teoreticky poměrně velký výběr, neboť nabídka databázových systémů na trhu je velmi široká a jednotlivé společnosti vyvíjejí stále nové verze svých produktů.

Nabízené produkty lze rozdělit na dvě skupiny. První typ – open source - je poskytován zdarma (včetně zdrojových kódů) a vyžaduje nulové nebo jen minimální náklady na pořízení a používání, což se tedy jeví jako zásadní výhoda. Druhým typem jsou proprietární, tedy komerční databázové systémy. Některé z nich jsou zdarma nabízeny v omezených startovacích verzích. Předpokládá se ale, že uživatelé přejdou později k vyšším (placeným) edicím.

Do skupiny open source databázových systémů bychom mohli zařadit MySQL, PostgreSQL nebo Firebird. Jako zástupce druhé skupiny bych uvedl databázový systém Oracle 11g společnosti Oracle, MS SQL server 2008 společnosti Microsoft a DB2 společnosti IBM.

Pro svou bakalářskou práci, kde porovnávám oba typy databázových systémů, jsem vybral z každé skupiny jednoho zástupce. Open source databáze budou zastoupeny databázovým systémem Firebird, jako zástupce komerční sféry byl vybrán databázový systém Oracle.

Tato bakalářská práce by mohla být základní pomůckou pro ty uživatele, kteří přemýšlejí, zda jim použití open source databáze s nulovými (nebo minimálními) náklady zabezpečí splnění jejich požadavků, nebo se rozhodují využít pro začátek některou z nižších edic komerčních databázových systémů a pak snadno (bez nutnosti zásahů do stávajících systémů) přejít k jejich vyšší verzi, která by komplexně řešila všechny problémy spojené s rostoucím objemem dat a dalšími uživatelskými nároky.

Práce je tedy určená především pro uživatele, kteří mají základní znalosti o databázích a hodlají si je rozšířit o informace, které jim poskytnou základ pro orientaci na trhu open source i komerčních databázových systémů. Řešení optimalizace SQL příkazů, ladění výkonu, detailní rozbor licenčních podmínek ani pohled na administrační úkony nebudou součástí této práce.

Hlavním cílem této práce bude:

- porovnat open source databázový systém Firebird s komerčním řešením společnosti Oracle
- poskytnout čtenáři informace potřebné pro rozhodování mezi open source a komerčním řešením.
- zjistit, zda může open source databáze konkurovat komerční, a to z hlediska definovaných oblastí a kritérií.
- prakticky ověřit výkon (dobu provádění SQL příkazů) obou srovnávaných databází

Mou snahou tedy bude najít a popsat jednak společné prvky a jednak unikátní vlastnosti a funkce obou databází v teoretické rovině a nakonec v praktické části ověřit porovnání výkonu obou databází v testovací aplikaci.

Pro splnění prvního a druhého cíle prostuduji tištěné a elektronické zdroje, které budou uvedeny na konci této práce. Poskytnou mi dostatek informací, které bude nutné analyzovat, utřídit a vhodným způsobem prezentovat. Oblasti a kritéria porovnávání budou sestavena na základě vlastních znalostí a zkušeností získaných studiem na vysoké škole a praxí v oboru databází. Praktické ověření výkonu proběhne v konzolové aplikaci vytvořené v jazyce Java.

Na začátku bude čtenář seznámen se základními pojmy, které se vyskytují v dalších částech práce a které považuji za důležité pro pochopení základní myšlenky práce. Následující část bude zaměřena na definici oblastí a kritérií, na jejichž základě budou obě databáze v teoretické části detailně porovnány.

Mou snahou bude najít unikátní vlastnosti a funkce obou databází v teoretické rovině. Nakonec bude v praktické části ověřeno porovnání výkonu obou databází v testovací aplikaci. Výsledky testování budou shrnuty do závěrečných tabulek.

2. Vymezení základních pojmů

Pro dobré pochopení následujících kapitol považuji za důležité nejdříve vysvětlit základní pojmy, které se týkají databází.

Data

Za data¹ považujeme jakékoliv fyzicky (materiálně) zaznamenané znalosti (vědomosti), poznatky, zkušenosti nebo výsledky pozorování procesu, projevu, činností a prvků reálného světa.

Databáze

Pod pojmem databáze (neboli datová základna)² bychom si měli představit data, které slouží více aplikacím. Cílem databázového systému je uspořádat datové zdroje na počítači tak, aby tyto zdroje mohly být efektivně používány.

Relační datový model

Relační datový model³ byl představen v 1970. Dodnes se ale používá pro ukládání dat v databázích. Model využívá k vyjádření dat a vztahů mezi nimi množinu tabulek. Každá tabulka má jedinečný název a je tvořena několika sloupci a řádky. Nejmenší hodnotou v relační struktuře je datová hodnota. Řádek tabulky reprezentuje vztah množiny hodnot. Tabulka je tedy množinou takových vztahů.

Systém řízení báze dat

Systém řízení báze dat⁴ je založen na relačním datovém modelu. Aby databázový systém mohl být označován jako relační (*Relation Database Management System*), musí splňovat minimálně tři následující pravidla:

¹ Buchalcevoá, A., Stanovská, I., Šimunek, M.: Základy softwarového inženýrství – základní témata, 1.vyd., skripta VŠE, 2002, ISBN 80-245-0346-8, s. 190

² Ibid

³ Ibid, s. 28

⁴ Ibid, s. 38

1. databáze se uživateli musí jevit jako množina tabulek
2. mezi tabulkami neexistují žádné uživatelem viditelné fyzické vazby
3. musí být k dispozici alespoň jeden uživatelsky přístupný jazyk pro manipulaci s daty, který umožní realizovat základní relační operace (projekce, selekce, kartézský součin, rozdíl a sjednocení), aniž by bylo nutné psát postup získání výsledku

Jazyk SQL

SQL (*Structure Query Language* – *strukturovaný dotazovací jazyk*)⁵ je neprocedurální jazyk využívající kombinace a relačního kalkulu matematické algebry. Byl ustanoven jako standard pro relační databázové jazyky – poslední verze je z roku 1992 (v dalších částech této práce také jako standard SQL92⁶).

SQL lze rozdělit do následujících oblastí⁷:

Jazyk pro definici dat (DDL – Data Definition Language)

Příkazy této skupiny začínají slovy *CREATE*, *ALTER*, *DROP* – používané pro vytváření, změnu nebo odstranění objektů⁸.

Jazyk pro manipulaci s daty (DML – Data Movement Language)

Do této skupiny patří příkazy *SELECT*, *UPDATE*, *DELETE* a *INSERT* – sloužící k výběru, aktualizaci, odstranění a vložení nových dat.

Jazyk pro řízení přístupových práv (DCL - Data Control Language) –

Tato skupina⁹ je zastoupena příkazy *GRANT* a *REVOKE* - vhodné pro přidělování nebo odebírání práv uživatelům.

⁵ Buchalceová, A., Stanovská, I., Šimunek, M.: Základy softwarového inženýrství – základní témata, 1.vyd., skript VŠE, 2002, ISBN 80-245-0346-8, s. 38

⁶ Standard SQL 92 [online]. Dostupné na:
<http://www.contrib.andrew.cmu.edu/~shadow/sql/sql1992.txt> (navštíveno 22. 10. 2009)

⁷ W3school – SQL Syntax [online]. Dostupné na:
http://www.w3schools.com/sql/sql_syntax.asp (navštíveno 2. 10. 2009)

⁸ Objektem zde může být databáze, tabulka, uložená procedura, index a další.

⁹ Z této skupiny se ještě vyčleňuje jazyk pro řízení transakcí (*TCL - Transaction Control Language*), který obsahuje příkazy *COMMIT* (pro potvrzení transakce) a *ROLLBACK* (pro zrušení transakce a návrat do původního stavu)

3. Stanovení oblastí a kritérií porovnávání

V této kapitole jsou popsány vybrané databázové systémy, použitý hardware, stanovené oblasti a kritéria porovnávání.

3.1. Porovnáváné databázové systémy

Do vlastního porovnávání jsem vybral Firebird 2.1.2 (Classic) a Oracle Database 10g Express Edition (dále také Oracle Express edice nebo pouze Oracle). Zejména kvůli použití lokálního jednouživatelského přístupu jsem při instalaci databázového systému Firebird vybral architekturu Classic (viz kapitola 4 - část Architektura). Oracle Express edice, která sdílí s ostatními edicemi stejné jádro databázového serveru, je (podle oficiálních stránek společnosti Oracle¹⁰) určena především pro studenty, začínající programátory nebo malé firmy, které ji také mohou použít i pro komerční účely.

3.1.1. Historie

Firebird

V roce 2000¹¹ firma Borland zdarma poskytla internetové komunitě vývojářů zdrojové kódy populárního databázového serveru InterBase 6.0 beta (pod názvem InterBase Open Edition a licencí IPL¹²). Souběžně ale pokračovala ve vývoji a prodeji komerční verze (tzn. bez poskytnutí zdrojových textů) této databáze. Krátce po uvolnění zdrojových kódů komunita založila nový samostatný projekt Firebird¹³, který dále pokračoval ve vývoji open source databáze.

¹⁰ Oficiální stránky společnosti Oracle – www.oracle.com

¹¹ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 3-4

¹² Licence IPL - InterBase Public License [online]. Dostupné na: <http://www.firebirdsql.org/index.php?op=doc&id=ipl> (navštíveno 1. 9. 2009)

¹³ Tento projekt nyní zaštiťuje nezisková organizace Firebird Foundation. Firebird Foundation [online]. Dostupné na: <http://www.firebirdsql.org/index.php?op=ffoundation>

Vývoj databáze InterBase Open Edition byl firmou Borland oficiálně ukončen v roce 2002 ve verzi 6.0.2. spolu s uvedením komerční verze InterBase 6.5. Z původní databáze InterBase 6.0 se tedy oddělily dvě hlavní vývojové větve – komerční verze InterBase firmy Borland a open source projekt Firebird, které spolu sdílejí 95% funkčnosti. V listopadu roku 2009 vyšla zatím poslední verze Firebird 2.5. Beta 2. Aktuální verze srovnávaného databázového systému je Firebird 2.1.2.

Oracle

Historie¹⁴ společnosti Oracle sahá do 70. let 20. století, kdy byl pod názvem Oracle založen databázový projekt pro americkou bezpečnostní službu (CIA). V roce 1979 vychází první komerční verze relačního databázového systému pod názvem Oracle Version 2¹⁵. V dalších verzích postupně přibývá transakční zpracování, zámky na úrovni řádků, programování v PL/SQL, rychlé zálohy, podpora jazyka Java¹⁶ a další funkcionality. V roce 2003 pak přichází první vydání databáze Oracle 10g – s novou architekturou *gridu* (česky mřížka nebo síť, která automaticky rozprostírá výpočetní sílu mezi více databázových serverů na základě počtu uživatelských požadavků). V září roku 2009 společnost vydává zatím poslední verzi databázi Oracle 11g Release 2.

Aktuální verze srovnávaného databázového systému je ale Oracle Database 10g Express Edition Release 2 (10.2.0.1). Další Oracle Express edice založená na verzi 11g se podle mého názoru objeví v druhé polovině roku 2010.

3.2. Použitý hardware

Všechny testy a praktická porovnání byly provedeny na osobním notebooku značky Compal HEL 80 s operačním systémem Microsoft Windows XP Professional CZ (Service Pack 3), základní deskou Mobile Intel Calistoga-PM i945PM osazenou procesorem Intel Core2 CPU T7200 s taktem 2000 MHz. Notebook obsahuje také paměť RAM o velikosti 2 GB DDR, grafický adaptér NVIDIA GeForce Go 7600 (256 MB) a harddisk SEAGATE ST9160821AS o velikosti 160 GB.

¹⁴ Oracle's 30th Anniversary (PDF) [online]. Dostupné na: http://www.oracle.com/oramag/profit/07-may/p27anniv_timeline.pdf (navštíveno 3. 10. 2009)

¹⁵ Oracle Version 1 napsaná v jazyce Assembler nikdy nebyla komerčně uvedena.

¹⁶ Oficiální stránky jazyka Java jsou dostupné na www.java.com.

3.3. Oblasti a kritéria porovnávání

Nejdříve jsem si stanovil oblasti porovnávání – cena a licence, funkcionality a technologie, architektura, administrace, instalace, řízení zálohy a obnovy dat, technická podpora a dokumentace a nakonec výkon. Do těchto oblastí jsem pak zařadil kritéria, podle nichž jsem v následující části oba databázové systémy teoreticky porovnal. Pro lepší představu a pochopení, co jednotlivé oblasti porovnávání přesně představují, považuji za důležité, poskytnout podrobný popis jednotlivých kritérií.

3.3.1. Cena a licence

Otázka ceny databáze a její licence často rozhoduje o výběru databáze, proto v tomto kritériu bude zohledněna nejen pořizovací cena databáze, ale i ceny jednotlivých uživatelských licencí.

3.3.2. Funkcionality a technologie

Po úvodním seznámení považuji za vhodné uvést funkční a technologickou stránku srovnávaných databázových systémů. Budu se tedy zabývat architekturou databázového serveru, uvedu konkrétní omezení z hlediska operačních systémů, hardwaru, velikosti databáze a poskytnutých rozhraní. Dále pak zhodnotím instalaci a uvedu administrační nástroje, které usnadňují a zjednodušují práci s datovými objekty databází.

Funkční a technologická kritéria jsou rozdělena do šesti skupin:

3.3.2.1. Omezení

Operační systém

Kritérium, které obsahuje výčet operačních systémů, na kterých lze danou databázi nainstalovat a provozovat, je velmi důležité, neboť může dramaticky omezit výběr databáze.

Hardware

Kritérium podporovaného hardwaru bude zachycovat technické aspekty výběru databáze jako například počet současně pracujících procesorů a velikost paměti RAM.

Velikost databáze

Již název kritéria napovídá, že v něm půjde především o maximální množství uložených dat v databázi.

Aplikační rozhraní

Toto kritérium shrnuje výčet poskytovaných aplikačních rozhraní tzv. API.¹⁷

3.3.2.2. Architektura

Předmětem zkoumání kritéria architektury bude detailní pohled a srovnání architektur obou databázových systémů.

3.3.2.3. Instalace

Kritérium instalace se bude zabývat především průběhem instalace a uživatelskou přívětivostí, velikostí instalace a instalačního balíku.

3.3.2.4. Administrace

Do kritéria o administraci jsem zahrnul nástroje pro správu databáze obsažené v instalaci. Zmíním také použité specializované aplikace pro administraci, jejich možnosti a funkce.

3.3.2.5. Datové typy

Cílem kritéria shrnujícího datové typy není prezentovat jejich úplný seznam, ale především ukázat rozdíly a omezení mezi srovnávanými databázemi a naznačit možnosti definice vlastních datových typů.

¹⁷ API – Application Programming Interface – aplikační rozhraní
Buchalceová, A., Stanovská, I., Šimunek, M.: Základy softwarového inženýrství – základní témata,
1.vyd., skript VŠE, 2002, ISBN 80-245-0346-8, s. 189

3.3.2.6. Databázové objekty

V kritériu databázových objektů opět neposkytnu jejich úplný výčet, ale pouze vyzdvihnu zvláštnosti a odlišnosti v jednotlivých databázích.

3.3.3. Konzistence

V části pojednávající o konzistenci se budu zabývat především transakčním zpracováním, řízením souběžného přístupu více uživatelů k datům a vzniku možných problémů s ním spojeným.

3.3.4. Řízení zálohy a obnovy dat

Mnoho společností závisí na svých životně důležitých datech, a proto vyžadují okamžitou dostupnost databází dvacet čtyři hodin denně, sedm dní v týdnu, tři sta šedesát pět dní v roce. S takto vysokými požadavky na dostupnost úzce souvisí i scénáře pro zálohování a obnovu dat.

V kritériu o zálohování a obnově dat budu řešit možnosti vytváření záloh, jejich různé typy (plná nebo přírůstková) a zda uživatelé mohou souběžně pracovat při zálohovacím procesu. Obnova dat představuje opačný proces k zálohování. Půjde tedy především o navrácení databáze do původního stavu po pádu serveru nebo uživatelské chybě.

3.3.5. Technická podpora a dokumentace

Hlavní náplní tohoto kritéria bude dostupnost technické podpory a množství oficiální i neoficiální dokumentace od výrobce.

3.3.6. Výkon

Toto kritérium představuje stěžejní část praktického ověření výkonu databází. Zachytím zde rychlosti prováděných SQL příkazů. Metodiku, postup testování a výsledky jsou uvedeny v kapitole 5 – Porovnání výkonu.

4. Porovnání na základě stanovených kritérií

V této části jsou porovnány postupně po jednotlivých oblastech a kritériích oba databázové systémy. Nejdříve je čtenáři představen historický vývoj, cena a licence, pod kterou se databáze dodává, dále jsou uvedeny limity a architektury obou databázových serverů. Následuje průběh instalace, možnosti administrace, konzistence transakcí a souběžný přístup k datům, mechanismy pro zálohování a obnovu, a dostupnost technické dokumentace.

4.1. Cena a licence

Nejdříve je čtenář seznámen s cenami jednotlivých databází a licencemi, pod kterými se databáze dodává.

Firebird

Přestože databáze Firebird¹⁸ je zdarma dostupná z oficiálních stránek organizace, její používání, změny zdrojového kódu i další distribuce jsou právně ošetřeny. To znamená, že za případné porušení licenčních podmínek hrozí právní postih. InterBase Open Edition a její zdrojový kód byl uvolněn pod licencí InterBase Public Licence (IPL)¹⁹, která je odvozená od Mozilla Public Licence verze 1.1²⁰.

Nové moduly Firebirdu jsou ale již vydávány pod licencí Initial Developer's Public Licence Version 1.0 (IDPL²¹).

¹⁸ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 4

¹⁹ InterBase Public Licence [online]. Dostupné na:
<http://www.firebirdsql.org/index.php?op=doc&id=ipl> (navštíveno 9. 9. 2009)

²⁰ Mozilla Public Licence [online]. Dostupné na:
<http://www.mozilla.org/MPL/MPL-1.1.html> (navštíveno 9. 9. 2009)

²¹ Initial Developer's Public Licence Version 1.0 [online].
Dostupné na: <http://www.firebirdsql.org/index.php?op=doc&id=idpl> (navštíveno 9. 9. 2009)

Oracle

Oracle Express Edition se na rozdíl od ostatních edic dodává zdarma²², ovšem za cenu toho, že se nejedná o plně funkční databázový systém - viz kapitola Funkcionalita a technologie – část Omezení. Na rozdíl od open source řešení, zde nejsou poskytnuty zdrojové kódy a nelze tedy do nich jakkoliv zasahovat. Na druhou stranu je zde pak snadný přechod na vyšší (ale komerční) verzi v případě růstu uživatelských nároků na databázi. Mohli bychom tedy tuto databázi z hlediska licencí považovat za „demoverzi“²³ placeného produktu společnosti Oracle.

Srovnání Express edice s ostatními edicemi

I když hlavním objektem porovnávání je Oracle Express edice, považuji za důležité alespoň nastínit základní srovnání s ostatními edicemi.²⁴

Všechny verze (včetně Express Edition) obsahují společné jádro databázového systému a navzájem se liší obsaženými komponentami, cenou a limity. (Aktuální ceník včetně cen licencí – viz příloha 1). Vyšší (placené) edice již mimo jiné obsahují plnou podporu 64 bitových procesorů, podporu operačního systému Unix, automatické zálohování a obnovu dat, administrační nástroje Oracle Enterprise Manager a podporu jazyka Java. Jejich velikost paměti RAM je omezena maximem operačního systému. Samozřejmě neobsahují žádný limit na množství uživatelských dat.

²² Oracle Database 10g Express Edition Downloads[online].
Dostupné na: <http://www.oracle.com/technology/software/products/database/xe/index.html>
(navštíveno 3. 10. 2009)

²³ Demoverze – verze komerčního produktu, která bývá omezená funkcionalitou (některé funkce nelze využít) nebo dobou používání (například po uplynutí 30 dní již nelze spustit)

²⁴ Edice Oracle Database 11g [online]. Dostupné na:
<http://www.oracle.com/global/cz/database/edice.html> (navštíveno 3. 10. 2009)

Oracle Database Lite

Trochu stranou od základních edic stojí Oracle Database Lite²⁵. Jedná se o komplexní řešení pro vývoj, implementaci a správu aplikací pro mobilní²⁶ a vestavěná (*embedded*) prostředí. Mobilní zařízení pak mohou být obousměrně synchronizována s podnikovou databází pomocí Oracle Database Lite Mobile Serveru.

4.1.1.1. Shrnutí

Oba databázové systémy jsou k dispozici zdarma, ale Oracle na rozdíl od Firebirdu neposkytuje zdrojové kódy a navíc Express edice obsahuje omezení, která jsou detailně popsána v následující kapitole.

4.2. Funkcionalita a technologie

Oblast funkcionality a technologií pojednává o databázových limitech, architektuře databázového serveru, průběhu instalace, možnostech administrace, datových typech a datových objektech. Na závěr jsou uvedeny jedinečné vlastnosti obou databázových systémů.

4.2.1. Omezení

V následující části je čtenáři poskytnut výčet operačních systémů, na které lze srovnávané databázové systémy nainstalovat a provozovat, dále pak omezení z hlediska hardwaru, maximální velikosti databáze a poskytnutých aplikačních rozhraní.

²⁵Oracle Database Lite 10g [online]. Dostupné na: http://www.oracle.com/database/lite_edition.html (navštíveno 3. 10. 2009)

²⁶ Tato edice byla využita v mobilních zařízeních značek Nokia a Motorola.

Firebird

Operační systém

Databáze Firebird²⁷ lze nainstalovat a spravovat na platformách Linux a Windows (32 i 64 bit verze), MAC OS, Solaris, HP-UX i různých variantách FreeBSD.

Hardware

Hardwarové²⁸ omezení se týká pouze architektury SuperServer, která neumí využít možnosti víceprocesorových serverů.

Velikost databáze

Velikost^{29;30} databáze není omezena (díky používání více databázových souborů). Limit na jeden databázový soubor závisí na použité platformě a souborovém systému. Například Windows 2000 (a novější) při použití souborového systému NTFS mají limit až 16 Exbibytů³¹.

Aplikační rozhraní

Databáze Firebird poskytuje aplikační rozhraní pro jazyky Delphi, C, C++, Java, PHP, Python, Perl i .NET, ale i rozhraní JDBC, ODBC a OLEDB.

²⁷ Firebird – Overview of Features [online]. Dostupné na: <http://firebirdsql.org/manual/ufb-about-features.html> (navštíveno 9. 9. 2009)

²⁸ Firebird – System Requirements [online]. Dostupné na: <http://firebirdsql.org/manual/ufb-about-sysreq.html> (navštíveno 9. 9. 2009)

²⁹ FAQ – What are the technical limits of Firebird? [online]. Dostupné na: <http://www.firebirdfaq.org/faq61/> (navštíveno 9. 9. 2009)

³⁰ FAQ - What's the limit on the Firebird database size? [online]. Dostupné na: <http://www.firebirdfaq.org/faq59/> (navštíveno 9. 9. 2009)

³¹ 16 Exbibytů odpovídá 17 miliardám Gigabytů.

Oracle

Operační systém

Oracle Express edici lze nainstalovat pouze na platformu Windows a Linux, a to pouze 32bitovou verzí.

Hardware

I na serverových sestavách s procesory osazenými více patičkami bude tato edice vždy využívat maximálně jednu z nich a bude ukrajovat 1GB paměti RAM.

Velikost databáze

Maximální velikost databáze byla stanovena na 5GB (4GB uživatelských dat + 1GB systém).

Aplikační rozhraní

Databáze poskytuje aplikační rozhraní³² pro jazyky C, C++, PHP, .NET, rozhraní ODBC³³, OLE DB a JDBC, ale na rozdíl od ostatních edic nepodporuje jazyk Java.

4.2.1.1. Shrnutí

Jediné omezení databáze Firebird spočívá v architektuře SuperServer (viz následující kapitola), která dokáže využít pouze jedno procesorové jádro.

Oracle Express edice je oproti Firebirdu podstatně omezenější, což jistě může ovlivnit výběr databázového systému. Nainstalovat ji lze pouze na operační systémy Windows nebo Linux (a navíc jen 32bitové verze). Pro svou výpočetní sílu bude vždy používat pouze jeden procesor a zabírat 1GB operační paměti. Maximální velikost databáze byla stanovena na 4GB uživatelských dat. A nakonec vůbec nepodporuje jazyk Java.

³² Oracle Database Express Edition 2 Day Developer Guide 10g [online]. Dostupné na: http://download.oracle.com/docs/cd/B25329_01/doc/appdev.102/b25108/xedev_intro.htm (navštíveno 3. 10. 2009)

³³ ODBC – Open Database Connectivity – standard datového rozhraní databází

4.2.2. Architektura

Tato část je zaměřena především na rozdíly mezi architekturou databázového systému Firebird a Oracle s jejich výhodami a nevýhodami.

Firebird

Multigenerační architektura

Databázový systém Firebird používá tzv. multigenerační architekturu³⁴, kterou programátoři poprvé implementovali již v databázovém systému InterBase. Stejné principy můžeme také nalézt v open source databázi PostgreSQL a částečně i v databázi Oracle.

Multigenerační architektura³⁵ stojí na dvou pilířích:

1. Vytváření více verzí záznamu – „verzování“

V databázovém systému Firebird se v průběhu transakce ukládá přímo do databáze více verzí stejného záznamu. (Tato skutečnost ovšem platí pouze pro sloupce, které jsou v průběhu transakce změněny.) Jednotlivé verze se od sebe odlišují číslem transakce, která ji vytvořila, a obsahují odkaz na předchozí verzi. Tvoří tak řetěz, kde aktuální verze záznamu je na jeho začátku.

2. Evidence stavu transakcí

Kromě čísla transakce a odkazu na minulé verze, obsahuje záznam i stav transakce. Firebird rozlišuje následující stavy: aktivní (*active*), připravený k potvrzení (*commit*), potvrzený (*committed*) a zamítnutý (*rollback*).

Pro lepší představu a správné pochopení vytváření verzí záznamů i evidence stavu transakcí jsem zachytil průběh jednoduché transakce do tabulky 1.

³⁴ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 53

³⁵ Ibid, s. 335

Tabulka 1- Průběh transakce

Číslo záznamu	STUDENT .JMENO	Číslo transakce	Předchozí transakce	Stav transakce
1		10		aktivní
1	R.Lelek	11	10	připravená k potvrzení
1	R.Lelek	12	11	zamítnutá
1		13	12	aktivní
1	Roman Lelek	14	13	připravená k potvrzení
1	Roman Lelek	15	14	potvrzená

Tato tabulka zachycuje průběh jednoduché transakce, která se snaží vložit v tabulce STUDENT do sloupce JMENO nejdříve hodnotu „R.Lelek“, která je ale v zápětí odvolána. Uživatel pak zadá správný tvar jména a potvrdí transakci příkazem COMMIT.

Kvůli vytváření několika verzí záznamů je nutné odstraňovat z databáze starší verze, které již nejsou potřeba. (V tabulce 1 to představují záznamy 10 až 14).

Automatický úklid databáze zajišťují³⁶:

Garbage Collection

Garbage Collection v průběhu načítání záznamu do paměti prochází řetěz předchozích verzí a odstraňuje takové, které už více nebudou potřeba.

Sweep – „zametání“

Protože Garbage Collection odstraní nepotřebné verze jen u načítaných záznamů, Sweep musí projít celý obsah databáze a odstranit všechny nepotřebné verze.

³⁶ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 222

Možnosti architektury serveru

Odlišná architektura serveru nijak neovlivňuje vnitřní strukturu databáze ani způsob práce klientů s databázovým serverem.

Firebird je k dispozici ve třech variantách:

1. Classic

Varianta Classic³⁷ představuje původní architekturu InterBase 6.0 (InterBase Open Edition). Pro každého připojeného uživatele se vytvoří nový serverový proces, který se stará výhradně o vyřizování požadavků tohoto uživatele.

2. SuperServer

Varianta SuperServer byla implementována firmou Borland při přenosu databáze InterBase na platformu Windows. Díky paralelnímu zpracování vláken umožňuje architektura SuperServer^{38;39} obsluhu více uživatelů současně.

3. Embedded

Speciální variantu Embedded^{40;41;42} využijí zejména vývojáři aplikací, kteří chtějí využít výhody transakčního zpracování a ukládání dat. Hodí se pro tvorbu jednouživatelských aplikací, demo verzí nebo pro vytváření katalogů na CD-ROM.

³⁷ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 17-18

³⁸ Ibid, s. 19-20

³⁹ The InterBase SuperServer Architecture [online]. Dostupné na:
http://www.ibphoenix.com/main.nfs?a=ibphoenix&page=ibp_ss_vs_classic (navštíveno 2. 9. 2009)

⁴⁰ Client and server combined: Firebird Embedded Server [online]. Dostupné na:
<http://www.firebirdsql.org/manual/ufb-cs-embedded.html> (navštíveno 1. 9. 2009)

⁴¹ Firebird Embedded a .NET – Jiří Činčura – Vyvojari.cz [online]. Dostupné na:
<http://www.vyvojari.cz/Articles/446-firebird-embedded-a-net.aspx> (navštíveno 2. 9. 2009)

⁴² Client and server combined: Firebird Embedded Server [online]. Dostupné na:
<http://www.firebirdsql.org/manual/fbmetasecur-embedded.html> (navštíveno 15. 9. 2009)

Autoři zde spojili možnosti architektury SuperServer s klientskou knihovnou tak, aby zachovali plnohodnotný databázový systém.

Mezi nedocenitelné výhody patří snadné použití ve vývojovém prostředí a jednoduchý přechod na ostatní typy architektury. To vše ale za cenu, že databáze není zabezpečená na úrovni operačního systému, neposkytuje žádné síťové připojení a může k ní přistupovat pouze jeden proces.

Oracle

Architekturu databázového systému Oracle můžeme rozčlenit do tří oblastí: fyzická, logická a architektura pamětí a procesů. **Fyzická oblast** představuje uložení vlastních uživatelských dat a zajišťuje správu databáze. Skládá se tedy z datových souborů (*data files*), kontrolních souborů (*control files*), souborů transakčního protokolu (*redo logs*) a dalších. **Logická oblast** určuje, jakým způsobem se bude používat fyzická oblast databáze. Skládá se z tabulkových prostorů, segmentů, rozsahů a bloků. **Architektura pamětí a procesů** propojuje fyzickou a logickou oblast a zajišťuje její konzistenci.

Fyzická oblast

Fyzická architektura představuje vlastní uložení uživatelských dat a správu databáze, ukládá se do několika typů souborů. Vybral jsem tři základní typy souborů: datové soubory (*data files*), kontrolní soubory (*control files*) a soubory transakčního protokolu (*redo logs*).

Datové soubory (data files)

Jak již název napovídá, datové soubory⁴³ představují vlastní fyzické uložení dat. Jsou to základní stavební kameny tabulkových prostorů (viz Logická architektura). Vlastní zápis ale zajišťuje proces, který běží na pozadí – DBW (viz tato kapitola).

⁴³ Kevin Loney, Marlene Theriault: Mistrovství v Oracle – Kompletní průvodce tvorbou, správou a údržbou databází, 1. vydání, nakladatelství Computer Press, 2002, ISBN 80-7226-635-7, s. 5 – 8

Soubory transakčního protokolu (Redo logs)

Do souborů transakčního zpracování (redo logů)⁴⁴ se ukládají průběhy všech transakcí. Každá databáze Oracle obsahuje alespoň tři aktivní soubory transakčního protokolu. Zápis do těchto souborů (viz proces LGWR) probíhá cyklicky – po zaplnění prvního redo logu, přichází na řadu druhý a nakonec třetí. Po zaplnění posledního souboru se data buď archivují anebo opět začne zápis do prvního souboru⁴⁵.

Kontrolní soubory (Control files)

Kontrolní soubory⁴⁶ obsahují informace o datových souborech, souborech transakčního protokolu a důležité informace o databázi (například její název, datum vytvoření, způsob posledního ukončení). Protože se jedná o klíčové soubory databáze, existují v několika kopiích na různých discích nebo serverech.

Logická oblast

Logická oblast určuje, jakým způsobem se bude používat fyzická oblast databáze. Skládá se tedy z tabulkových prostorů, segmentů, rozsahů a bloků.

Tabulkový prostor (Tablespace)

Základní jednotku dělení logické architektury databáze Oracle představuje tzv. tabulkový prostor.^{47;48} Každá databáze obsahuje minimálně jeden tabulkový prostor – SYSTEM. (Automaticky jsou ještě vytvořeny tabulkové prostory SYSAUX, USERS a UNDO). Tabulkový prostor může patřit pouze do jediné databáze. Obdobně platí, že každý databázový objekt musí patřit právě do jednoho tabulkového prostoru.

⁴⁴ Kevin Loney, Marlene Theriault: Mistrovství v Oracle – Kompletní průvodce tvorbou, správou a údržbou databází, 1. vydání, nakladatelství Computer Press, 2002, ISBN 80-7226-635-7, s. 5 – 8

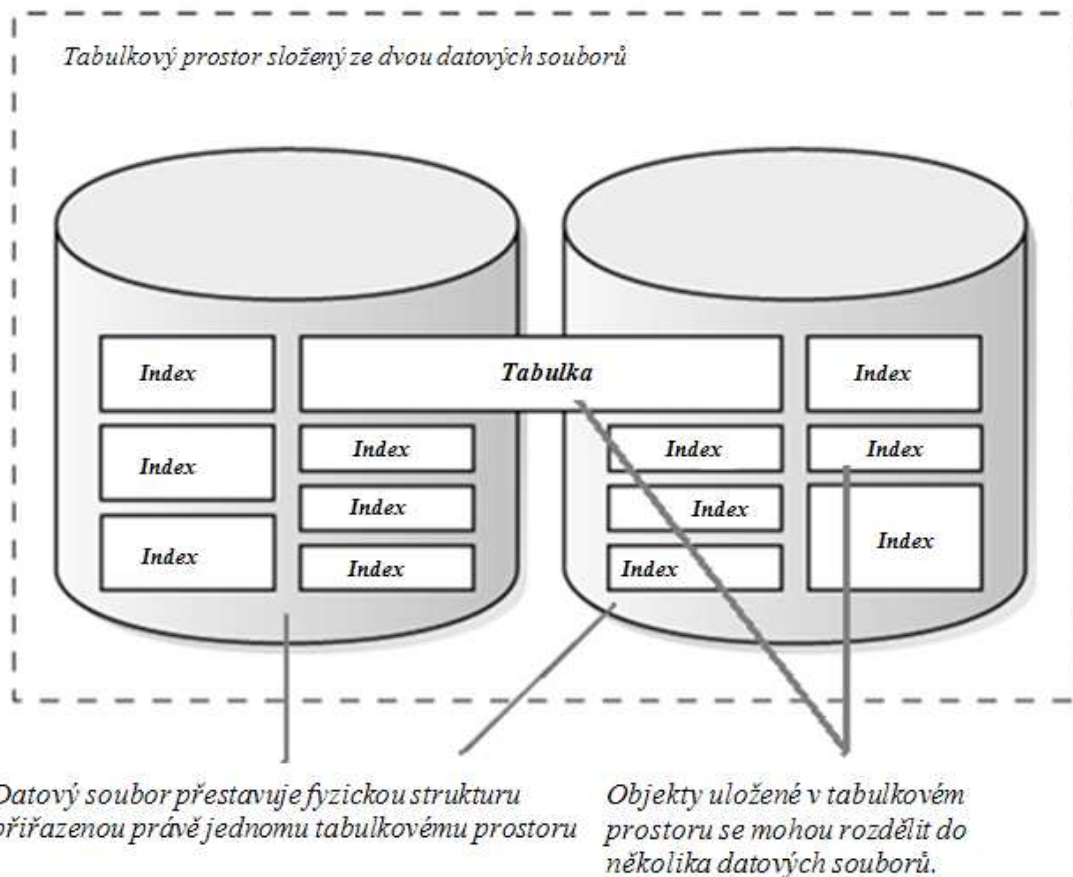
⁴⁵ Soubory transakčního protokolu jsou zálohovány v případě, že databáze je spuštěna v módu ARCHIVELOG.

⁴⁶ Kevin Loney, Marlene Theriault: Mistrovství v Oracle – Kompletní průvodce tvorbou, správou a údržbou databází, 1. vydání, nakladatelství Computer Press, 2002, ISBN 80-7226-635-7, s. 5 – 8

⁴⁷ Ibid

⁴⁸ Oracle Database Concepts 10g - Tablespaces, Datafiles, and Control Files [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/physical.htm#i2006 (navštíveno 11. 11. 2009)

Objekt (například tabulka) ale může být rozdělen do více databázových souborů (*data files*) Obrázek 1 názorně zobrazuje rozdělení tabulkového prostoru na dva datové soubory, ve kterých je uložena tabulka.



Obrázek 1 - Tabulkový prostor složený ze dvou datových souborů, zdroj:⁴⁹

Obrázek zobrazuje tabulkový prostor rozdělný do dvou datových souborů, které obsahují databázové objekty (tabulky a indexy).

Datové bloky (Data Blocks) a rozsahy (Extents)

Nejmenší jednotku uložení dat v databázi Oracle představuje jeden datový blok⁵⁰, jehož velikost odpovídá násobku velikosti bloku operačního systému. Databáze Oracle používá a uvolňuje místo právě v podobě datových bloků. Za rozsah se považuje několik souvislých bloků systému Oracle.

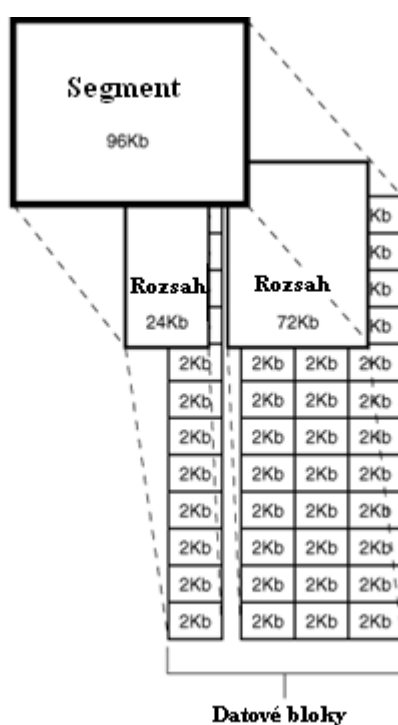
⁴⁹ Oracle Database Concepts 10G - Datafiles and Tablespaces [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/img/cncpt037.gif

⁵⁰ Oracle Database Concepts 10G - Introduction to the Oracle Database [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/intro.htm#i60798 (navštíveno 9. 11. 2009)

Segmenty (Segments)

Segment⁵¹ se skládá ze všech rozsahů, které byly přiděleny pro uložení konkrétní datové struktury v tabulkovém prostoru. Například data tabulky jsou uložena ve svém vlastním „*tabulkovém*“ segmentu, zatímco indexy jsou uloženy ve svém vlastním „*indexovém*“ segmentu. Pokud segment již nemá volné místo, databáze Oracle automaticky přiřadí další volný rozsah.

Vztahy mezi datovými bloky, rozsahy a segmenty zobrazuje obrázek 2. Rozdělení na tabulkové a indexové segmenty rovněž dokumentuje obrázek 1.



Obrázek 2 - Vztah datových bloků, rozsahů a segmentů, zdroj:⁵²

Segment (Segment), se skládá ze dvou (a více) rozsahů (Extents). Rozsahy se dále mohou dělit na datové bloky (Data Blocks).

⁵¹ Kevin Loney, Marlene Theriault: Mistrovství v Oracle – Kompletní průvodce tvorbou, správou a údržbou databází, 1. vydání, nakladatelství Computer Press, 2002, ISBN 80-7226-635-7, s. 45

⁵² The Relationships Among Segments, Extents, and Data Blocks [online]. Dostupné na: http://download-uk.oracle.com/docs/cd/B19306_01/server.102/b14220/img/cncpt027.gif

Oblast paměti a procesů

Architektura paměti zajišťuje vztahy mezi fyzickou a logickou oblastí. Můžeme ji ještě rozdělit na dvě části: systémovou globální paměť (*System Global Area*) a programová globální paměť (*Program Global Area*).

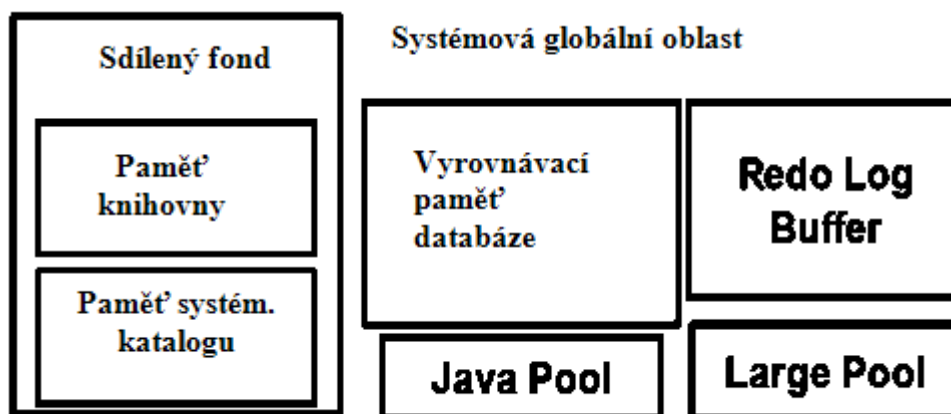
Systémová globální oblast (System Global Area - SGA)

Systémová globální oblast⁵³ (viz obrázek 3) je (spolu s procesy na pozadí) základním paměťovým prvkem instance. Instanci⁵⁴ bychom si mohli představit jako „přístup“ do databáze. Pokud k databázi přistupuje více uživatelů, rozdělí se systémová globální oblast mezi ně, proto se také někdy nazývá „sdílená“.

Mezi její nejdůležitější součásti patří vyrovnávací paměť databáze - *Database Buffer Cache*, která obsahuje kopie datových bloků z datových souborů a zvyšuje tím výkon databáze při vybírání a aktualizaci dat, sdílený fond - *Shared Pool*, který obsahuje poslední spouštěné SQL příkazy, a dočasné paměti pro soubory redo log - *Redo log Buffer*, které obsahují záznamy pro rekonstrukci (*undo*) nebo znovu provedení změn (*redo*). Fond jazyka Java - *Java Pool* a rozsáhlý fond - *Large Pool* představují volitelné součásti této oblasti.

⁵³ Oracle Database Concepts 10G - Oracle Memory Architecture [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/memory.htm#g30926 (navštíveno 11. 11. 2009)

⁵⁴ Dalším „skrytým“ omezením Oracle Express edice je možnost spustit právě jednu instanci databáze.



Obrázek 3 – Komponenty systémové globální paměti, zdroj:⁵⁵

Systémovou globální paměť můžeme rozdělit na sdílený fond (Shared Pool), který se ještě dělí na vyrovnávací paměť knihovny (Library Cache) a vyrovnávací paměť systémového katalogu (Data Dictionary Cache). Systémová globální paměť dále obsahuje vyrovnávací paměť databáze (Database Buffer Cache), fond jazyka Java (Java Pool), rozsáhlý fond (Large Pool) a vyrovnávací paměť pro soubory transakčního protokolu (Redo log buffer).

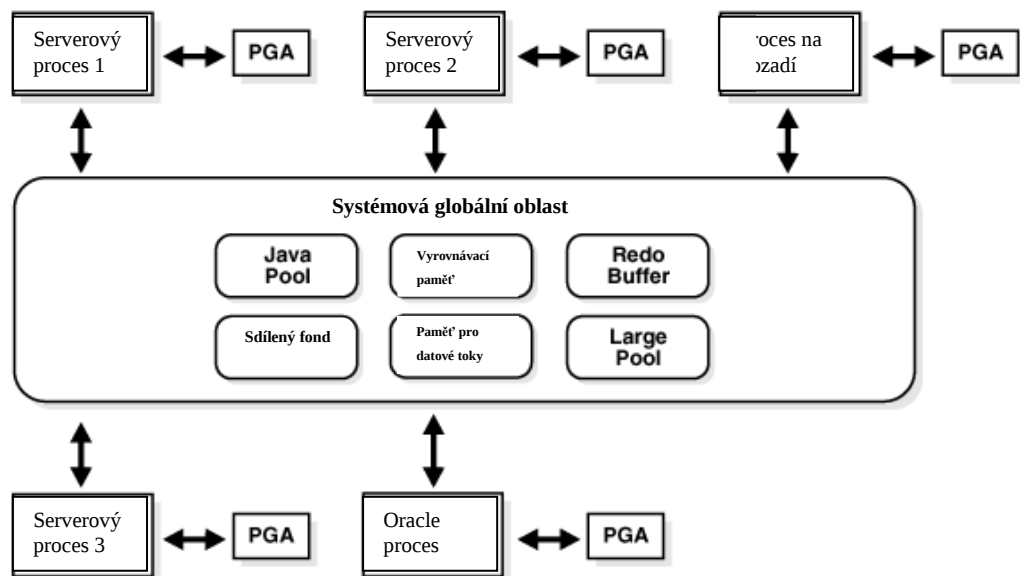
Programová globální oblast (Program Global Area - PGA)

V databázi Oracle probíhají procesy uživatelské (začínající přípojevacím požadavkem), serverové (začínající v okamžikem spojení) a procesy na pozadí (začínající současně s instancí databáze). Programová globální oblast^{56;57} představuje paměť rezervovanou pro každý proces. Obrázek 4 zobrazuje pohled na programovou i systémovou globální oblast.

⁵⁵ Oracle Architectural Components [online]. Dostupné na: <http://www.oracle.com/technology/tech/migration/isv/docs/OracleArchitectureOverview.pdf> (navštíveno 21. 11. 2009)

⁵⁶ Oracle Database Concepts 10G - Introduction to the Oracle Database [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/intro.htm#i60798 (navštíveno 11. 11. 2009)

⁵⁷ Oracle Database Concepts 10G - Oracle Memory Architecture [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/memory.htm#g30926 (navštíveno 11. 11. 2009)



Obrázek 4 – Programová globální oblast a systémová globální oblast, zdroj:⁵⁸

Uprostřed leží systémová globální oblast (System Global Area), která se skládá z fondu jazyka Java (Java Pool), sdíleného fondu (Shared Pool), vyrovnávací paměť databáze (Buffer Cache), paměti pro datové toky (Streams Pool), rozsáhlého fondu (Large Pool) a vyrovnávací paměti pro soubory transakčního protokolu (Redo buffer). Okolo leží procesy (serverové, databázové a procesy běžící na pozadí), pro které je alokována vlastní programová globální oblast.

Procesy na pozadí

Procesy na pozadí⁵⁹ zajišťují konzistentní vztahy mezi fyzickou a logickou architekturou. Jako nejdůležitější bych uvedl procesy DBW (Database writer, který provádí zápis uživatelských dat ze sdílené paměti databáze (Database Buffer Cache) přímo do datových souborů a LGWR (Log writer), který je zodpovědný za zapisování nových položek z vyrovnávací paměti pro transakční protokol (redo log buffer) do jeho souborů (redo log souborů).

⁵⁸ Oracle Database Concepts 10G - Oracle Memory Architecture [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/img/cncpt151.gif (navštíveno 11. 11. 2009)

⁵⁹ Oracle Database Concepts 10G - Oracle Process Architecture[online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/process.htm#i16977 (navštíveno 11. 11. 2009)

4.2.2.1. Shrnutí

Databázový systém Firebird používá multigenerační architekturu založenou na evidenci stavu transakcí a vytváření více verzí záznamů přímo v databázi. Nepotřebné verze odstraní procesy Garbage Collection a Sweep. Architektura je k dispozici ve verzi Classic, SuperServer a Embedded. Verzi *Classic* je výhodnější použít v jednouživatelských aplikacích a v aplikacích, které běží na stejném počítači (serveru) jako databázový server. Verze *SuperServer* naopak vítězí na jednoprosesorových počítačích (serverech), kde do databáze přistupuje více uživatelů. Verze *Embedded* je vhodná pro vytváření jednouživatelských aplikací, demo verzí nebo aplikací využívající výhody transakčního zpracování a ukládání dat.

Architektura databázového systému Oracle se skládá ze tří oblastí fyzické, logické a paměťové. Fyzická oblast představuje vlastní uložení uživatelských dat, logickou si můžeme představit jako systém, jakým se uživatelská data organizují uvnitř databáze a paměťová se stará o propojení těchto dvou oblastí. Tato oblast například zapisuje uživatelská data tabulky (logická oblast) do datových souborů (fyzická oblast).

V databázi Oracle se ukládají informace (uživatelská data, přístup k databázi apod.) do několika typů souborů (datové soubory, kontrolní soubory, soubory transakčního protokolu). Oproti tomu Firebird uchovává veškeré informace v jednom souboru.

Výhoda architektury databázového systému Firebird spočívá v neblokování zápisu při čtení dat. To zaručuje větší propustnost než systémy se zámkou a transakčním protokolem, který používá databázový systém Oracle Express edice. Na druhou stranu, tyto systémy díky transakčnímu protokolu mají k dispozici přírůstkové zálohování.

Naopak za nevýhodu databázového systému Firebird bychom mohli považovat zpomalení provádění uživatelských požadavků způsobené vytváření více verzí záznamů - zejména u čtení transakcí, které byly jinou transakcí změněny nebo odstraněny.

Výhody i nevýhody architektur typu Classic a SuperServer jsem shrnul do tabulky 2.

Tabulka 2- Srovnání typů architektur Classic a SuperServer^{60; 61}

Classic	SuperServer
Výhody	
pád jedné instance serveru neovlivní ostatní uživatele	menší režie
možnost zabudování do mobilních zařízení	průměrně lepší výkon při velkém počtu současně pracujících uživatelů
možnost přímého přístupu k databázovým souborům	sdílení vyrovnávací paměti
vhodné pro aplikace v jednouživatelském režimu	
od verze FB 1.5 i na platformě Windows	
Nevýhody	
nízká úroveň sdílení informací mezi procesy	nevyužívá plně výpočetní schopnosti počítačů s více procesory

4.2.3. Instalace

Tato oblast pojednává o vlastní instalaci databázového systému a o nárocích na diskový prostor.

Firebird

Velikost instalačního balíčku je překvapivě malá – necelých 7 MB, velikost nainstalovaného databázového systému pak zabírá skoro 21 MB.

⁶⁰ Classic and SuperServer architectures [online]. Dostupné na:
<http://firebirdsql.org/manual/ufb-about-arch.html> (navštíveno 1. 9. 2009)

⁶¹ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 282

Vlastní instalace probíhá v uživatelsky příjemném prostředí, ve kterém jsou již základní hodnoty nastaveny. Uživatel tedy vybere pouze jazyk instalace, odsouhlasí licenční podmínky, zadá umístění na svém disku a vybere typ požadované architektury. Na platformě Windows má ještě na výběr z variant spouštění databázového serveru – buď jako aplikaci anebo jako službu.

Oracle

Velikost instalačního balíčku je oproti Firebirdu mnohonásobně větší, přesně 157 MB, stejně tak i celková velikost databázového systému se zvedla skoro na 1 GB.

Instalace rovněž probíhá v uživatelsky příjemném prostředí, ve kterém jsou již základní hodnoty nastaveny. Uživatel tedy vybere pouze jazyk instalace, odsouhlasí licenční podmínky a vybere umístění na svém disku.

4.2.3.1. Shrnutí

Databázový systém Firebird požaduje oproti Oracle Express edici pouze minimální nároky na volné místo disku.

Instalace obou databázových systémů probíhala jednoduše, tedy bez nastavování pokročilejších parametrů. Jen u databáze Firebird jsem musel zvážit požadovaný typ architektury a možnost spouštění databázového serveru.

4.2.4. Administrace

V této oblasti je čtenář seznámen s výčtem nástrojů pro správu databáze obsažené v instalaci a jsou zde také zmíněny použité specializované aplikace pro administraci, jejich možnosti a funkce.

Firebird

Databáze Firebird je dodávána s několika nástroji – *GBAK* pro zálohování, *ISQL* pro spouštění SQL příkazů, *GSTAT* pro analýzy a statistiky a *GFIX* pro opravu poškozené databáze. Jejich použití je však podle mého názoru pro běžné uživatele příliš složité, protože neposkytují žádné grafické rozhraní, ale jen příkazový řádek.

Na oficiálních stránkách komunity⁶² je k dispozici poměrně velké množství volně stažitelných i placených aplikací, které se liší obsaženými funkcemi.

Osobně jsem vyzkoušel SQL Manager 2008 Lite⁶³, který ale bohužel neobsahoval možnost zálohování a obnovy databáze. Tuto funkcionalitu obsahuje pouze placená verze tohoto programu. Využil jsem ale nástroje pro tvorbu nových databází, jejich správu, nástroje pro vytváření různých databázových objektů jako jsou tabulky, domény, pohledy, uložené procedury, uživatelsky definované funkce, spouště, výjimky, generátory a role.

Za velmi užitečnou funkci považuji možnost extrahovat celou databázi (včetně dat i metadat) do jediného skriptu, který by pak mohl fungovat jako určitá forma zálohování.

Oracle

Oracle již po instalaci obsahuje webové prostředí, které lze rozdělit do čtyř částí. První slouží pro administraci uživatelů, systémové paměti a místa, v druhé části můžeme prohlížet databázové objekty, třetí nám pomáhá při exportu a importu dat a čtvrtou část využijeme pro spouštění SQL příkazů.

Nicméně pro větší uživatelskou pohodlnost jsem pro administraci používal aplikaci Oracle SQL Developer 1.5⁶⁴, který poskytuje detailní pohled na databázové objekty jako například tabulky, pohledy, materializované pohledy, indexy, databázové linky, funkce, procedury, spouště, sekvence, synonyma i XML schémata. Obsahuje také debugger pro ladění SQL skriptů.

Za velký přínos považuji dostupnost modulů pro správu verzí (CVS⁶⁵, Subversion⁶⁶), modul pro migraci databáze (např. do Microsoft Access) a další nástroje pro kopírování, export a porovnání databází.

⁶² Oficiální stránky komunity Firebird - <http://firebirdsql.org/>

⁶³ EMS SQL Manager for InterBase/Firebird [online]. Dostupné na: <http://sqlmanager.net/en/products/ibfb/manager>

⁶⁴ Oracle SQL Developer 1.5 – [online]. Dostupné na: <http://www.oracle.com/technology/software/products/sql/index.html>

⁶⁵ CVS (Concurrent Version System) - systém, který slouží ke správě verzí projektu.

⁶⁶ Subversion (zkráceně SVN) je systém pro správu a verzování zdrojových kódů, náhrada za starší CVS.

4.2.4.1. Shrnutí

Oba databázové systémy obsahují základní administrační nástroje obsluhovatelné přes příkazový řádek, ale jejich použití je pro běžné uživatele příliš složité, využijí ho spíše zkušení administrátoři. Důležité administrační funkce zpřístupňují specializované aplikace - pro Firebird jsem použil SQL Manager 2008 Lite a pro databázi Oracle SQL Developer. Oba tyto nástroje mi poskytly přístup a detailní nastavení všech důležitých databázových objektů. SQL Manager navíc obsahuje možnost extrakce všech dat i metadat v jediném SQL skriptu. Oracle SQL Developer navíc obsahuje prostředky pro ladění a moduly pro týmovou spolupráci, migraci, kopírování a export dat.

4.2.5. Datové typy

V této části probíhá porovnání datových typů obsažených v obou databázových systémech s datovými typy vyžadovanými standardem SQL92. Jsou zde také uvedeny možnosti definování vlastních datových typů.

Firebird

Datové typy⁶⁷ databáze Firebird odpovídají datovým typům ve standardu SQL92⁶⁸, ale navíc databáze Firebird nabízí datový typ BIGINT (pro uložení 64bitových celých čísel⁶⁹) a datový typ BLOB pro uložení rozsáhlých dat s proměnlivou délkou (např. obrázky, texty, videa).

Datový typ BLOB ještě můžeme rozdělit na podtypy – podtyp s kladným nebo záporným celým číslem, které blíže specifikuje strukturu dat. Hodnota nula je standardně rezervovaná pro libovolná binární data, hodnota 1 pro text a záporné hodnoty pro uživatelské podtypy.

⁶⁷ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 282

⁶⁸ Mezi datové typy definované standardem SQL92 patří: CHARACTER, CHARACTER VARYING, BIT, BIT VARYING, NUMERIC, DECIMAL, INTEGER, SMALLINT, FLOAT, REAL, DOUBLE PRECISION, DATE, TIME, TIMESTAMP a INTERVAL.

⁶⁹ Svou velikostí tedy odpovídá datovému typu NUMERIC(18,0)

Pomocí tohoto podtypu můžeme odlišit zpracování stejného typu uživatelských dat uložených v odlišných formátech (např. obrázky ve formátu BMP a JPG).

Definice vlastních datových typů - domény

Domény slouží k vytvoření vlastního datového typu pomocí jednoho nebo více datových typů poskytnutých databázovým systémem. Tuto funkcionalitu využijeme například pro vytvoření jednorozměrného pole⁷⁰ s kontrolou vložených hodnot – viz následující SQL příkaz.

```
CREATE DOMAIN KATEGORIE_ZAKAZNIKU AS VARCHAR(10)
CHARACTER SET WIN1250
CHECK (VALUE IN ('A', 'B', 'C'));
```

Tento SQL kód vytvoří doménu (pole), která obsahuje pouze 3 znaky – „A“, „B“ a „C“ v kódování WIN1250

Oracle

Oracle mapuje datové typy⁷¹ standardu SQL92 podle následujících pravidel: pro datové typy CHAR a CHARACTER Oracle používá datový typ CHAR, pro datové typy CHARACTER VARYING a CHAR VARYING pak VARCHAR2. Číselné datové typy NUMERIC, DECIMAL, DEC, INTEGER, INT a SMALL INT odpovídají datovému typu NUMBER. V rámci PL/SQL Oracle podporuje místo standardních datových typů REAL a DOUBLE PRECISION datový typ FLOAT.

Oracle poskytuje ještě navíc následující datové typy: NVARCHAR2, INTERVAL day to second, INTERVAL year to month, TIMESTAMP with time zone

⁷⁰ Lze definovat i vícerozměrná pole.

⁷¹ Kevin Loney, Marlene Theriault: Mistrovství v Oracle – Kompletní průvodce tvorbou, správou a údržbou databází, 1. vydání, nakladatelství Computer Press, 2002, ISBN 80-7226-635-7, s. 32 – 33

(ukládá datum a čas včetně příslušného nastavení časového pásma, a to buď jako posun vůči centrálnímu času, anebo jako název oblasti), TIMESTAMP with local time (má podobnou funkci jako předchozí datový typ, ale při ukládání je datum přizpůsobeno časovému pásmu databáze a při načítání časovému pásmu klienta.), RAW (vhodný pro ukládání binárních dat), BLOB (vhodný zejména pro ukládání velkého rozsahu binárních dat – až 4GB), CLOB (vhodný pro ukládání textů – až 4GB), NCLOB (pro znakové objekty v národních jazycích) a BFILE, který představuje cestu k externímu binární souboru (např. k obrázku), jehož velikost je omezena operačním systémem.

Definice vlastních datových typů – „sběrné“ typy (Collection Types)

V Oracle Express edici jsou k dispozici dvě možnosti definice vlastních datových typů⁷². První spočívá ve vytvoření uspořádané množiny (pole), která obsahuje určitý počet stejných datových typů. (Například kolekce pěti textových řetězců). Druhá možnost spočívá ve vytvoření vnořené tabulky. Její prvky nejsou upořádané, ale stejně jako v předchozí možnosti obsahují stejný datový typ.

4.2.5.1. Shrnutí

Oba databázové systémy poskytují základní datové typy dané standardem SQL92 (viz poznámka pod čarou 68).

Firebird navíc ještě poskytuje numerický datový typ BIGINT a datový typ BLOB.

Oracle je oproti Firebirdu bohatší na datové typy. Poskytuje navíc „doplňkové“⁷³ datové typy pro ukládání dat v národních jazycích, intervalové datové typy pro ukládání časového období, rozšíření časových značek o časové pásmo, datové typy RAW a BLOB pro ukládání binárních dat, CLOB a NCLOB pro znakové objekty a BFILE pro externí binární soubory.

⁷² Oracle Database Concepts 10G -Object Datatypes and Object Views [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/objects.htm#g446482 (navštíveno 11. 11. 2009)

⁷³ Myslím tím datové typy začínající písmenem „N“ (NVARCHAR, NVARCHAR2, NCLOB apod.). Umožňují tak ukládat národní znakové sady – např. čínské znaky.

Oba databázové systémy také poskytují možnosti definice vlastních datových typů. Databázový systém Firebird nabízí domény, Oracle je řeší pomocí sběrných typů.

4.2.6. Databázové objekty

Výčet všech databázových objektů by byl jistě nad rozsah této práce, proto jsou zde uvedeny pouze rozdílné objekty z obou databázových systémů.

Firebird

Databáze Firebird⁷⁴ používá pro tvorbu uložených procedur a spouští jazyk *Procedural SQL (PSQL)*, který představuje podmnožinu jazyka SQL doplněnou o rozšíření zaměřená na procedurální zpracování dat. Obsahuje tedy lokální proměnné a příkazy přiřazení hodnoty, příkazy pro řízení běhu programu (*IF*, *THEN*, *ELSE*, *WHILE...DO*), kontextové proměnné⁷⁵ představující původní (*OLD*) a nové (*NEW*) hodnoty sloupců tabulek, zasílání uživatelsky definovaných událostí, výjimky a volání uložených procedur.

Uložené procedury (Stored procedures)

Uloženou proceduru⁷⁶ si můžeme představit jako databázový objekt, který vykonává uživatelem definovanou funkcionalitu (například ve školní databázi přesune všechny studenty s ukončeným studiem z tabulky *Studenti* do tabulky *Absolventi*). Její volání probíhá pomocí následujícího příkazu:

EXECUTE <název procedury> (parametry)

⁷⁴ Pavel Císař: *InterBase/Firebird – Tvorba, administrace a programování databází*, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 303

⁷⁵ Kontextové proměnné jsou dostupné pouze pro spouště.

⁷⁶ Pavel Císař: *InterBase/Firebird – Tvorba, administrace a programování databází*, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 304 - 310

Firebird kromě výše uvedené funkcionality nabízí i tzv. „dotazovatelné uložené procedury“. Díky jazyku PSQL, který umožňuje procházet výsledky uložené procedury, lze spustit i následující kód:

```
SELECT * FROM <název procedury> (parametry).
```

Databázové spouště (Triggers)

Databázové spouště⁷⁷ fungují obdobně jako „standardní“⁷⁸ spouště, ale na úrovni celé databáze. Jejich akce tedy začínají při připojení (odpojení) od databáze nebo při zahájení, potvrzení nebo odvolání celé transakce.

Uživatelsky definované výjimky (Exceptions)

Uložené procedury mohou přerušit svůj průběh kvůli chybě a svůj stav signalizovat pomocí uživatelských výjimek⁷⁹. Jedná se o objekty v databázi, které mohou používat uložené procedury nebo spouště.

Vytvoření a vyvolání uživatelsky definované výjimky probíhá následujícím SQL kódem:

```
CREATE EXCEPTION TEST80 'Hodnota A je větší než  
hodnota B!';
```

Vytvoření výjimky TEST, která obsahuje text Hodnota A je větší než hodnota B!

```
IF (A > B) THEN EXCEPTION TEST
```

Vyvolání výjimky pomocí tohoto SQL kódu (za předpokladu splnění podmínky A>B) dostaneme text definovaný ve výše uvedeném SQL příkazu.

⁷⁷ Release Notes Firebird 2.1 [online]. Dostupné na:
<http://firebirdsql.org/rlsnotesh/rlsnotes210.html#rnfb210-dbtriggers> (navštíveno 11. 11. 2009)

⁷⁸ Spouště představují funkcionalitu automaticky spouštěnou před (nebo po) vložení, aktualizaci nebo mazání řádku tabulky (nebo pohledu). V praktické části této práce byly použity při definici primárních klíčů.

⁷⁹ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 316

⁸⁰ Obecně lze zapsat vytvoření výjimky jako CREATE EXCEPTION jméno ‘text chybové hlášky’

Uživatelsky definované funkce (UDF - User Defined Functions)

Databáze Firebird standardně neobsahuje velké množství funkcí, ale funkcionalitu lze rozšířit o uživatelsky definované (externí) funkce⁸¹. Mohou být vytvořené v libovolném programovacím jazyce (většinou se jedná o programovací jazyk C), který je schopen vytvořit dynamicky linkované knihovny funkcí, a pro jejich volání bude používat konvenci *stdcall*⁸².

Jednotlivé funkce musí být před použitím nainstalovány na databázový server a registrovány v každé databázi, kde je uživatel hodlá používat. Instalace probíhá nakopírováním zkompilované binární knihovny (např. *my_UDF_function.dll*) do adresáře UDF v instalační složce databáze Firebird a spuštěním registračního skriptu (např. *my_UDF_function.sql*). (viz následující SQL kód). Na webových stránkách komunity Firebird jsou k dispozici již hotové funkce, které lze zdarma stáhnout, upravovat i používat. Existují funkce pro pokročilejší práci s řetězcí, operace s čísly (například pro generování „náhodných“ čísel), operace s daty nebo konverzní funkce (například pro převod mezi teplotními stupnicemi) a jiné.

```
DECLARE EXTERNAL FUNCTION my_UDF_function
    RETURNS <datový typ> BY VALUE
    ENTRY_POINT 'my_UDF' MODULE_NAME my_UDF_function.dll;
```

Tento SQL skript zaregistruje a připraví k použití externě definovanou funkci obsaženou v knihovně my_UDF_function.dll

⁸¹ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 328 – 329

⁸² Stdcall - Standard calling convention for the Microsoft Win32 API x86 calling conventions

Oracle

Pohledy (Views) a materializované pohledy (Materialized Views)

Pohledy ^{83;84} bychom si nejlépe mohli představit jako virtuální tabulky, které mají definované sloupce z jedné (a více) tabulek (nebo pohledů). Na rozdíl od tabulek však pohledy neobsahují žádná data. Mezi hlavní funkce pohledů patří prezentace vybrané části dat. Například zdrojová tabulka se skládá z 50 sloupců a definováním pohledu můžeme zobrazit pouze prvních deset. Pohledy také poskytují možnost zabezpečení přístupu k datům – můžeme stanovit přesný počet sloupců (řádků), které uživatel uvidí.

Materializovaný pohled obsahuje vypočítané, sumarizované nebo replikované výsledky. Na rozdíl od pohledů jsou v nich uložena i uživatelská data. Materializované pohledy se používají v datových skladech, kde je třeba přistupovat k agregovaným hodnotám.

Synonyma (Synonyms)

Pod pojmem synonymum bychom si měli představit alternativní název pro databázové objekty např. pro tabulky, pohledy, sekvence, balíčky nebo uložené procedury. Synonyma můžeme rozdělit do dvou skupin – uživatelská (*private*) a veřejná (*public*). Pokud vytvoříme dvě stejně pojmenovaná synonyma patřící ale do různých skupin, přednost má uživatelské (*private*).

⁸³ Oracle Database Concepts 10G - Schema Objects [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/schema.htm#i20690 (navštíveno 12. 11. 2009)

⁸⁴ Kevin Loney, Marlene Theriault: Mistrovství v Oracle – Kompletní průvodce tvorbou, správou a údržbou databází, 1. vydání, nakladatelství Computer Press, 2002, ISBN 80-7226-635-7, s. 40

Funkce (Function)

Uložené funkce jsou podobně jako uložené procedury bloky PL/SQL⁸⁵ kódu. Liší se ale tím, že po zavolání funkce vždy dostaneme výsledek, uložené procedury ale vrátit nic nemusí. Oracle na rozdíl od Firebirdu obsahuje velké množství funkcí již po instalaci.

Balíčky (Package)

Skupina uložených procedur a funkcí uložených jako logický celek v databázi představuje balíček⁸⁶. Jejich výhody spočívají především v lepší organizaci, snadnějším zabezpečení a zlepšení výkonu (načtení celého balíčku je rychlejší oproti postupnému načítání jednotlivých procedur).

Spouště (Triggers)

Oracle kromě „standardních spouští“ (viz poznámka pod čarou č. 78) nabízí i spouště⁸⁷, které jsou svázány se systémovými nebo uživatelskými událostmi. Jedná se především o spuštění nebo ukončení databáze a přihlášení nebo odhlášení uživatele.

4.2.6.1. Shrnutí

Databázový systém Firebird nám na rozdíl od Oracle Express Edition umožňuje (díky jazyku PSQL) vytvořit „dotazovatelnou“ uloženou proceduru, což je v databázi Oracle nemožné. Lze také definovat spoušť, reagující na začátek, potvrzení nebo odmítnutí transakce. Na rozdíl od Oracle Express edice, nabízí také možnost definice vlastních výjimek, které lze přiřazovat uloženým procedurám nebo spouštím. Databázový systém Firebird v základní instalaci bohužel neobsahuje velké množství funkcí, a proto poskytuje definici uživatelských funkcí.

⁸⁵ Jazyk SQL obohacený o procedurální prvky: lokální proměnné a příkazy přiřazení hodnoty, příkazy pro řízení běhu programu, volání uložených procedur apod.

⁸⁶ Oracle Database Concepts 10G - SQL, PL/SQL, and Java [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/sqlplsql.htm#sthref3652 (navštíveno 16. 11. 2009)

⁸⁷ Oracle Database Concepts 10G - Triggers [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/triggers.htm#i10900 (navštíveno 16. 11. 2009)

Databázový systém Oracle Express Edition nabízí pro uložení agregovaných výsledků materializované pohledy, které Firebird neobsahuje. Další unikátní objekty, mezi které patří synonyma, která ulehčují pomocí alternativního názvu přístup k databázovým objektům, balíčky pro snazší organizaci funkcí a uložených procedur, jsou k dispozici také jen v Oracle Express Edition. Akce spouští lze mimo jiné spojit i s přihlášením nebo odhlášením uživatele.

4.2.7. Zvláštní funkcionalita

Tato část pojednává o specifických vlastnostech obou porovnávaných databází, které se vyskytují pouze v jedné z nich.

Firebird

Fyracle

Už z názvu vyplývá, že se jedná o spojení databázového systému Firebird s řešením Oracle. Konkrétně se jedná o zvláštní Oracle mód databáze Firebird. Pomocí Fyracle^{88;89} lze minimalizovat migrační náklady při přechodu z komerčního software (Oracle) na open source (Firebird). Největší problémy bývají především s uloženými procedurami, spouštěmi a použitými datovými typy. Souvisí to s rozdílnou implementací standardu SQL92. Tuto „přenositelnost“ kódu Oracle (PL/SQL) na kód Firebirdu (PSQL) Fyracle ošetřil implementací speciálního PL/SQL kompilátoru (překladače).

⁸⁸ Fyracle: Oracle-mode Firebird – [online]. Dostupné na:
http://www.janus-software.com/fb_fyracle.html (navštíveno 28. 9. 2009)

⁸⁹ Fyracle: Oracle-mode Firebird – [online]. Dostupné na:
<http://www.janus-software.com/fyracle.pdf> (navštíveno 28. 9. 2009)

Stínování databáze (Shadowing)

Databázový systém Firebird obsahuje unikátní funkcionalitu nazvanou „stínování“⁹⁰ – *shadowing*. Vytvořením stínu databáze (například na jiném serveru) vytvoříme reálnou kopii databáze a zajistíme okamžité automatické obnovení v případě pádu databázového serveru. Pro stínování není třeba dalšího softwaru, vše zajišťuje přímo databázový server. Vlastní proces stínování probíhá po každé ukončené transakci. V případě pádu hlavní databáze (stínované) lze přesměrovat požadavky klientských aplikací do vytvořeného stínu do té doby, dokud se databáze (stínovaná) neopraví a neobnoví.

Oracle

Oracle Application Express

Oracle Application Express je vlastní vývojové prostředí zabudované přímo do administračního modulu⁹¹ Oracle Express Edition. Slouží především pro tvorbu webových aplikací využívající databázi Oracle. Mezi velké výhody určitě patří skutečnost, že není nutná znalost programování, protože vlastní vývoj probíhá pouze pomocí průvodců a konfiguračních obrazovek. Oracle Application Express také obsahuje nástroje pro správu uživatelů a emailového klienta.

Technologie Flashback (Flashback Technology)

Flashback⁹² technologii obsahují všechny edice databázového systému Oracle. Jedná se o funkcionalitu, která rychle navrací poškozenou databázi (nejčastěji po uživatelské chybě) do původního stavu. Použitím této technologie můžeme například prohlížet a opravit data odstraněná před deseti minutami (na místo dlouhého obnovování celé databáze).

⁹⁰ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 216

⁹¹ Tlačítko pro vstup do Oracle Application Express se nezobrazuje systémovým administrátorům.

⁹² Oracle Flashback Technology [online]. Dostupné na:
http://www.oracle.com/technology/deploy/availability/htdocs/Flashback_Overview.htm
(navštíveno 10. 10. 2009)

V Oracle Express edici byla tato funkcionality použita komponentou Flashback Query⁹³, která nám umožňuje pokládat dotazy na podobu dat v minulosti.

4.2.7.1. Shrnutí

Stínování databáze a mód Fyrcle u Firebirdu, stejně tak jako technologie Flashback a Oracle Application Express u databázového systému Oracle se řadí mezi unikátní funkcionality, a proto se nacházejí pouze v jedné databázi.

Fyrcle představuje přechod od databázového systému Oracle k open sourceovému řešení Firebird. Tento speciální mód byl jako jediný úspěšně použit v aplikaci Compiere ERP+CRM⁹⁴.

Vytvořením stínu databáze můžeme zabezpečit dostupnost databáze Firebird pro klientské aplikace i v případě pádu databázového serveru. Stínování databáze ale nedokáže ochránit před uživatelskými chybami, ty jsou přeneseny i na stín.

Oracle technologie Flashback předpokládá, že za většinu chyb mohou databázoví uživatelé, a proto považuje za důležité, rychle a jednoduše se z nich zotavovat. Přichází tedy s revoluční technologií, pomocí které si opět můžeme zobrazit data, která jsme změnili před deseti minutami a opravit je. Tato technologie také šetří čas, protože není nutné obnovovat celou databázi.

⁹³ Ostatní edice ještě mohou obsahovat Flashback Database, Flashback Table, Flashback Drop, Flashback Versions Query a Flashback Transaction Query. Vždy se jedná o návrat do minulosti na určité úrovni (příkaz, transakce, tabulka, databáze).

⁹⁴ Compiere ERP+CRM – Open source ERP (řízení a plánování zdrojů) a CRM (řízení vztahů se zákazníky) řešení od společnosti Compiere (oficiální stránky dostupné na: <http://www.compiere.com/>)

4.3. Konzistence

V části pojednávající o konzistenci půjde především o transakční zpracování, řízení souběžného přístupu více uživatelů k datům a s ním spojené problémy.

Databázoví uživatelé (jejich aplikace) nepracují přímo se soubory databáze, ale komunikují s databázovým serverem pomocí tzv. transakcí. Pod pojmem „transakce“ bychom si měli představit uspořádaný blok příkazů, které mají zabezpečit konzistenci dat. Každá transakce má svůj začátek, dále pak tělo, ve kterém jsou napsány příkazy a také má svůj konec. Příkladem jednoduché transakce je následující SQL kód:

```
BEGIN
SELECT * FROM STUDENT
INSERT INTO STUDENT (JMENO) VALUES „ROMAN“
COMMIT
```

Příkazem BEGIN definujeme začátek transakce, následuje tělo obsahující výběr všech studentů z tabulky STUDENT a vložení nového studenta. Vše je ukončeno (potvrzeno) příkazem COMMIT.

Souběžný přístup k datům

Při přístupu více uživatelů k datům mohou vzniknout problémy způsobené tím, že více uživatelů v jednom okamžiku přistupují ke stejným záznamům. Jedná se o nereprodukovatelné čtení, fantómové řádky a „špinavé“ čtení⁹⁵.

Problém nereprodukovatelného čtení (Non-repeatable read)

K tomuto problému dochází tehdy, když data přečtená první transakcí jsou změněna druhou transakcí. Při opakovaném čtení stejných řádků (opakování první transakce) jsou načteny jiné hodnoty než při původním čtení.

⁹⁵ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 48 - 49

Pro lepší pochopení uvádím názorný příklad: V tabulce STUDENT je ve sloupci JMÉNO hodnota Roman. První uživatel v prvním čtení zobrazí všechny záznamy z tabulky STUDENT a zobrazí se mu výsledek „Roman“. V druhé transakci současně probíhá příkaz od druhého uživatele, který mění studenta „Roman“ na studenta „Karel“. Při druhém čtení první uživatel uvidí pouze studenta „Karel“. Původní hodnotu („Roman“) tedy nelze znovu zobrazit.

Problém fantómových řádků (Phantom rows)

Tento problém je podobný předchozímu, ale dochází k němu nikoliv změnou řádků, ale jejich přidáváním. Při opakování transakce tedy dostaneme kromě původních i nově přidané.

Pro lepší pochopení pokračuji v předchozím příkladě: Výchozí situace je stejná, ale druhý uživatel místo upravování, přidává nového studenta „Pavel“. V druhém čtení tedy první uživatel uvidí kromě studenta „Roman“ i studenta „Pavel“. Kromě původních hodnot se tedy „zázračně objevila“ i nová data.

Problém špinavého čtení (Dirty read)

Tento problém nastává tehdy, pokud se v souběžně běžící transakci čtou nepotvrzené změny z jiných transakcí.

Pro lepší pochopení opět uvedu názorný příklad: V původní tabulce STUDENT jsou ve sloupcích STUDENT, DLUH hodnoty Roman a 0. První uživatel zapíše dluh 100, ale transakci nepotvrdí (nezadá příkaz *commit*). Současně druhý uživatel zjišťuje dlužníky a jejich částky, kde se mu zobrazí výsledek Roman, 100. Ale první uživatel zjistí, že udělal chybu a odvolá svou transakci (použije příkaz *rollback*) a následně opraví dluženou částku zpět na 10. Druhý uživatel ale opravu dlužné částky už nezjistí.

Úroveň izolace transakcí (Transaction Isolation Level)

Databázový server se snaží řídit požadavky uživatelů tak, aby nedocházelo k výše uvedeným problémům. Úroveň izolace můžeme považovat za vlastnost transakce. Platí také, že transakce obsahující stejné bloky příkazů mohou mít různé úrovně izolace.

Standard SQL92 definuje následující izolační úrovně:

Read Uncommitted

V této úrovni jsou souběžné aktualizace blokovány, ale transakce mohou číst i dosud nepotvrzené změny provedené jinými transakcemi. (Problém „špinavého“ čtení).

Read Committed

Tato úroveň, stejně jako předchozí, blokuje souběžné aktualizace, ale navíc blokuje i čtení nepotvrzených změn jiných transakcí. I v této úrovni však může docházet k problémům s nereprodukovatelným čtením a fantómovými řádky.

Repeatable Read

Tato úroveň v sobě obsahuje vlastnosti předchozí úrovně a navíc zajišťuje pohled na čtená data tak, aby jednou přečtená data nebyla již změněna. Při přidávání nových dat však může vzniknout problém s fantómovými řádky.

Serializable

Tato úroveň izolace zabraňuje nejen neopakovatelnému čtení, ale i fantómovým řádkům. Transakce se tedy chovají tak, jako by byly prováděny postupně, a nikoliv souběžně. Ostatní transakce mohou číst zpracovávaná data, ale nemohou je měnit ani přidávat.

Firebird

Autoři InterBase⁹⁶ již dříve počítali s tím, že k jedné databázi bude přistupovat více uživatelů. Jejich množství není nijak omezeno, závisí především na platformě a konfiguraci hardwaru. Klienti mohou také pracovat s více databázemi současně. Tyto operace databáze Firebird zabezpečuje díky dvoufázovému potvrzovacímu protokolu.

⁹⁶ Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1, s. 48 - 50

Firebird poskytuje tři základní úrovně izolace: *Read Committed*, jejíž vlastnosti odpovídají stejnojmenné úrovni uvedené ve standardu SQL92. Dále pak úroveň *Snapshot*, kde je zaručen pohled na data v rozsahu úrovně *Serializable* standardu SQL92, ale nezaručuje, že data nebyla nebo nebudou v průběhu transakce změněna jinou transakcí. A v poslední řadě úroveň *Snapshot table stability*, která již zaručuje, že přečtená data nemohou být v průběhu transakce změněna jinou transakcí. Tato garance je ovšem za cenu blokace celé tabulky a to u všech tabulek zpracovávaných v průběhu této transakce.

Pro implementaci transakcí byly využity možnosti multigenerační architektury (vytváření více verzí řádků a evidence stavu transakce). Při čtení záznamu databázový systém vyhledá první verzi řádku vytvořenou potvrzenou transakcí (v izolační úrovni *Read Committed*) nebo potvrzenou transakcí s nižším číslem (v izolační úrovni *Repeatable Read*). Při změně záznamu (řádku) je situace o trochu složitější. Jestliže již existuje verze řádku s číslem transakce, které je vyšší, než přísluší aktuální transakci, mohou nastat dvě situace:

a) V případě, že tuto verzi vytvořila potvrzená transakce, nelze řádek aktualizovat a dojde ke kolizi transakcí. (Potvrzené změny tedy nelze aktualizovat).

b) Jestliže byla tato verze záznamu vytvořena nepotvrzenou transakcí, databázový server v závislosti na izolační úrovni buď ohlásí chybu, nebo pozastaví zápis do ukončení příslušné transakce. Pokud byla transakce odvolána, může být provedena aktualizace. Jestliže transakce byla potvrzena, databázový server ohlásí chybu.

Verze záznamů, které byly vytvořeny odvolanými nebo nepotvrzenými transakcemi se v obou případech vynechávají/přeskakují/vypouštějí.

Kolize transakcí při souběžné aktualizaci záznamů lze zjistit přímo ze seznamu verzí daného řádku.

Oracle

Oracle poskytuje izolační úrovně dle standardu SQL92 *Read Committed* a *Serializable*. Navíc ještě je k dispozici úroveň *Read-only*, která odpovídá *Repeatable Read* nebo *Serializable*, ale bez aktualizace dat. Transakce tak vidí pouze ty změny, které byly potvrzeny do začátku transakce.

Databázový server Oracle zabezpečuje současný přístup k datům, zabraňuje aktualizaci shodných dat v jednom okamžiku a minimalizuje čekací dobu na dokončení práce ostatních uživatelů díky principu tzv. uzamykání (zámků) dat. Při běžné práci databázový systém uzamkává data na úrovni řádků. Můžeme rozlišovat dva typy zámků - výlučné a sdílené.

Sdílené zámky

Sdílený zámek se použije při aktualizaci řádku, pokud ve stejném okamžiku neprobíhá druhá transakce, která by se snažila aktualizovat daný řádek.

Výlučné zámky

Výlučný zámek se použije, pokud se ve stejném okamžiku pokouší stejný řádek aktualizovat dvě transakce. Platí zde zásada, že první bude obsloužen uživatel, který jako první žádal o aktualizaci.

Multi-Version Read Consistency – „Inteligentní správa konkurenčního přístupu“

Jedná se o mechanismus, který zajišťuje, aby transakce, která data čte, neblokovala transakci, která právě data zapisuje. Multi-Version Read Consistency⁹⁷ bychom mohli rozdělit do dvou oblastí: *Statement level consistency* zajišťující čtení dat dotazu tak, že všechna data pocházejí ze stejného časového okamžiku. Druhou součástí pak tvoří *Transaction level consistency* zajišťující konzistenci čtení dat pro všechny dotazy v transakci.

⁹⁷ Oracle Database Concepts 10G - Data Concurrency and Consistency [online]. Dostupné na: http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/consist.htm#sthref1944 (navštíveno 11. 11. 2009)

4.3.1.1. Shrnutí

Při přístupu více uživatelů ke stejným datům mohou nastat problémy nereprodukovatelného čtení (v druhé transakci současně probíhá aktualizace), mohou se objevit fantómové řádky (v druhé transakci souběžně probíhá přidávání dat) nebo tzv. špinavé čtení (při čtení nepotvrzených transakcí). Databázový server se snaží těmto problémům předcházet různou úrovní izolace transakcí.

Firebird poskytuje úrovně *Read Committed*, *Snapshot* a *Snapshot table stability*.

Na rozdíl Oracle Express edice ale nepoužívá standardně zámky, jediné pokud to vyžaduje vyšší stupeň izolace transakce.

Databázový systém Oracle vychází ze standardu SQL92 a poskytuje tedy úrovně *Read Committed* a *Serializable*, navíc ale také nabízí úroveň *Read-Only*.

4.4. Řízení zálohy a obnovy dat

Mnoho společností kriticky závisí na životně důležitých datech, a proto vyžadují okamžitou dostupnost databází dvacet čtyři hodin denně, sedm dní v týdnu, tři sta šedesát pět dní v roce. S takto náročnými požadavky na dostupnost úzce souvisí i scénáře pro zálohování a obnovu dat.

Firebird

Instalace databázového serveru již obsahuje utility pro zálohování i obnovu dat – *gbak* a *nbackup*, ale bohužel neobsahují grafické uživatelské rozhraní, pouze příkazový řádek. Pro zálohování databáze Firebird je také k dispozici poměrně veliké množství volně stažitelných i placených programů.

Osobně jsem vyzkoušel aplikaci *Firebird Database Backup/Restore V1.0*⁹⁸, kde je třeba nastavit základní údaje o zálohované databázi jako host (*local host*), port (3050), uživatelské jméno (*sysdba*), heslo (*masterkey*), cestu k zálohované databázi (*D:\Firebird_2_1\TESTDB.FDB*) a cestu pro budoucí zálohu (*D:\Zaloha.gbk*).

⁹⁸ Firebird Database Backup/Restore V1.0 [online]. Dostupné na:
http://download.cnet.com/FireBird-Backup-Restore/3000-2242_4-10688595.html

Zálohovat lze za plného provozu databáze (zálohování probíhá jako transakce) a existuje také možnost zálohovat pouhým kopírováním souboru databáze (např. *TESTDB.FDB*) na jiný disk.

Vygenerování SQL skriptu (např. z administračního nástroje SQL Manager) obsahující všechna data i metadata by se také dalo považovat za formu zálohování.

Přírůstkové zálohování je k dispozici až od verze Firebird 2.0 a to právě díky utilitě *nbackup*.

Pro obnovení databáze jsem opět použil aplikaci *Firebird Database Backup/Restore V1.0*, kde jsem pouze upravil cestu k zdrojovému souboru, který jsem vytvořil v předchozí části (*D:\Zaloha.gbk*), cílovou databázi (*D:\Firebird_2_1\TESTDB-restore.FDB*) a nakonec vybral možnost *Restore*. Databázi Firebird nelze obnovit do již běžící databáze.

Oracle

Oracle obsahuje utilitu pro zálohování – *RMAN (Recovery Manager)*, ale jeho použití probíhá stejně jako u Firebirdu přes příkazový řádek. Již při instalaci se ale v hlavní nabídce Start, ve složce Oracle Database 10 G Express Edition vytvoří zástupce „Backup Database“, který spustí dávkový soubor *Backup.bat*. Aktivuje se tak utilita *RMAN* s již definovaným výchozím nastavením.

Obnova probíhá stejně lehce jako zálohování, jen s tím rozdílem, že v nabídce Start spustíme dávkový soubor „*Restore.bat*“. Výpis z průběhu obnovy – viz příloha 6. Oracle má v sobě obsaženou komponentu Koš (*Recycle Bin*), do které jsou umísťovány smazané objekty, ze kterého může být provedena jejich jednoduchá obnova.

4.4.1.1. Srovnání

Pro zálohování a obnovu dat v základní instalaci oba databázové systémy poskytují utility dostupné přes příkazový řádek.

Databáze Firebird lze na rozdíl od Oracle Express Edition zálohovat pouhým kopírováním souboru databáze nebo vygenerováním SQL skriptu obsahující data. Zálohovací aplikace poskytují grafické rozhraní, které zpříjemní a urychlí práci.

Pro zálohování databáze Oracle je k dispozici pouze utilita Recovery Manager a s instalací dodávané dávkové soubory, které ji spustí s předem nastavenými parametry. Lze zálohovat i za plného provozu databáze.

4.5. Technická podpora a dokumentace

V této části čtenáře seznámím s dostupností technické podpory a dokumentace od výrobce srovnávaných databázových systémů.

Firebird

K dispozici existuje poměrně velké množství zdarma dostupných informačních zdrojů a to i v několika světových jazycích.

Již s instalací databázového systému dostáváme základní dokumentaci (novinky v této verzi (*Release Notes*), opravené chyby (*Bug Fixes*), instalační návod (*Installation Guide* apod.), dále pak i odkaz na oficiální stránky komunity a informace o rozšíření jazyka SQL.

Za nejvýznamnější pramen považuji oficiální stránky projektu Firebird⁹⁹, sekce „*Documentation*“. Je zde na výběr od základní dokumentace (tzv. *Quick Start Guide* a *FAQ – Frequently Asked Questions* – často kladené otázky), přes uživatelské manuály (např. jak nainstalovat databázi Firebird na platformu Linux; zálohování a migrace dat) a tzv. *Release Notes* neboli poznámky k nově vydaným verzím, až po tzv. *Quick Sheets* (stručné výtahy nejdůležitějších vlastností, funkcí apod.). Nechybí zde ani a úplné znění licenčních podmínek. Kvůli velkému množství často kladených otázek (*FAQ*) komunita vytvořila zvláštní portál¹⁰⁰ rozdělený do kategorií podle typu dotazu.

⁹⁹ Firebird Documentation Index [online]. Dostupné na:
<http://firebirdsql.org/index.php?op=doc> (navštíveno 15. 9. 2009)

¹⁰⁰ The Firebird FAQ [online]. Dostupné na:
<http://www.firebirdfaq.org/> (navštíveno 15. 9. 2009)

Za další důležitý informační zdroj považuji tzv. diskusní skupiny – např. Google Groups¹⁰¹ a nebo Yahoo! Groups¹⁰², kde na otázky odpovídají kromě ostatních uživatelů i programátoři Firebirdu.

Nakonec bych neměl opomenout dostupnou dokumentaci od firmy IBPhoenix¹⁰³ a pomoc od komunity *SourceForge.net* dostupnou online na oficiálních stránkách.¹⁰⁴

Oracle

Za další „omezení“ Oracle Express edice by se dala považovat neexistence oficiální technické podpory, podpory po telefonu nebo aktualizací tohoto produktu. Mezi základní informační zdroje tedy patří v instalaci dodávaný návod „Jak začít používat databázi Oracle?“ (*Oracle Database Express Edition Getting Started Guide*). Dále pak je cenným zdrojem informací oficiální fórum Oracle Express Edition¹⁰⁵, a především *Oracle Database Documentation Library*¹⁰⁶. Jedná se o online¹⁰⁷ knihovnu manuálů obsahující návody rozdělené do několika sekcí: „instalační“ (např. pro instalaci databáze na platformě Linux nebo pro upgrade na vyšší verzi), „administrační“ (např. „*Jak se během 2 dnů stát databázovým administrátorem*“), „vývojářská“ (obsahuje manuály pro různá vývojová prostředí) a konečně „licenční“ obsahující plné znění licenčních podmínek.

¹⁰¹ Google Groups – Firebird (CZ) [online]. Dostupné na:
http://groups.google.com/group/firebird_cz (navštíveno 15. 9. 2009)

¹⁰² Yahoo! Groups - Support for Users of Firebird Releases [online]. Dostupné na:
<http://tech.groups.yahoo.com/group/firebird-support/messages> (navštíveno 15. 9. 2009)

¹⁰³ IBPhoenix Documentation [online]. Dostupné na:
http://www.ibphoenix.com/main.nfs?a=ibphoenix&page=ibp_document (navštíveno 15. 9. 2009)

¹⁰⁴ Sourceforge.net [online]. Dostupné na:
http://sourceforge.net/support/getsupport.php?group_id=9028 (navštíveno 15. 9. 2009)

¹⁰⁵ Oracle Database 10g Express Edition [online]. Dostupné na:
<http://forums.oracle.com/forums/forum.jspa?forumID=251> (navštíveno 10. 10. 2009)

¹⁰⁶ Oracle Documentation Library [online]. Dostupné na:
<http://www.oracle.com/pls/xe102/homepage> (navštíveno 10. 10. 2009)

¹⁰⁷ Dokumentaci lze i stáhnout ve formátu HTML nebo PDF.

Pro všechny edice je k dispozici fórum Toma Kyta¹⁰⁸ (viceprezident společnosti Oracle).

4.5.1.1. Shrnutí

Pro databázi Firebird je k dispozici velký výběr informačních zdrojů. Základní dokumentaci obsahuje již vlastní instalace databázového serveru. K dispozici, najednou a volně k stažení je její největší část umístěna na oficiálních stránkách komunity Firebird. Nalezneme zde základní dokumentaci, FAQ, novinky v aktuální verzi, i stručné výtahy nejzajímavějších částí a samozřejmě plné znění licenčních podmínek.

Protože databázový systém Oracle je dodáván zdarma, neposkytuje společnost Oracle žádnou technickou podporu, aktualizace ani pomoc na telefonu. Mezi hlavní informační zdroje tedy patří speciálně vytvořené fórum Oracle Express Edition a Oracle Database Documentation Library.

¹⁰⁸ Ask Tom Kyte [online]. Dostupné na:
<http://asktom.oracle.com/pls/asktom/f?p=100:1:0> (navštíveno 10. 10. 2009)

5. Porovnání výkonu

Tato část představuje stěžejní část praktické části mé bakalářské práce. Snažil jsem se do ní zachytit přípravu testování a vlastní postup testování. Nakonec je prezentováno výsledné porovnávání rychlosti prováděných příkazů v databázovém serveru.

5.1. Příprava testování

Nejdříve jsem definoval a připravil tabulky, které jsem pomocí generátoru naplnil náhodnými testovacími daty. Dále jsem definoval skupiny SQL příkazů a jejich možnosti rozšíření, které jsem použil v testovací aplikaci vytvořené v jazyce Java.

Vytvoření tabulek

V obou databázích jsem pomocí administračních nástrojů vytvořil deset tabulek pojmenovaných TABLE_T01 až TABLE_T10. Každá obsahuje průměrně dvacet sloupců, ve kterých jsem v různém počtu použil datové typy VARCHAR (s maximálním počtem pět, deset, dvacet, padesát a sto znaků), DATE, INTEGER, DOUBLE (DOUBLE PRECISION) a TIMESTAMP. První sloupec každé tabulky se nazývá PK a obsahuje primární klíč, který má datový typ INTEGER a je před vložením nové hodnoty generován.

Naplnění tabulek

Pro naplnění všech tabulek deseti tisíci řádky jsem použil volně dostupnou aplikaci Datagenerator 0.35¹⁰⁹. Bylo však nutné definovat omezující pravidla pro každý datový typ. Pro generování náhodných hodnot datového typu VARCHAR budou použita pouze písmena malé české abecedy bez háčeků (tzn. a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z).

¹⁰⁹ Datagenerator 0.35[online]. Dostupné na:
<http://sourceforge.net/projects/datagenerator/>

Pro číselný typ INTEGER aplikace náhodně vybrala hodnotu z intervalu (0 až 2147483647¹¹⁰), obdobně datové typy DOUBLE PRECISION a DOUBLE jsem omezil intervalem (0 až 999999999). Použitá aplikace bohužel neposkytovala možnost vygenerovat záporná čísla ani čísla s desetinnou čárkou pro datový typ DOUBLE. Pro generování časových údajů (datum a datum včetně času) jsem musel použít SQL příkazy¹¹¹, které vybírají aktuální datum ze systémových tabulek databáze.

Skupiny SQL příkazů

V této části definuji kategorie SQL příkazů, které budu následně spouštět ve vytvořené aplikaci.

Pro otestování výkonu srovnávaných databází jsem vybral následující skupiny (kategorie) SQL příkazů, které se snaží simulovat reálné business procesy (konkrétní znění SQL příkazů viz příloha 2):

Kategorie výběrů dat obsahuje: výběr dat obsahující jednoduchou podmínku (WHERE C1 = náhodné číslo), výběr dat obsahující agregaci (WHERE AVG (C3>náhodné číslo), výběr dat s použitím jednoho (a více) příkazu JOIN, výběr dat s příkazem CROSS JOIN (a více příkazy JOIN), výběr dat obsahující vnořený výběr dat, spojení výběrů dat pomocí příkazu UNION.

Kategorie operace s daty obsahuje: aktualizace sloupce pomocí příkazu UPDATE, vložení dat pomocí příkazu INSERT a smazání dat použitím příkazu DELETE.

Kategorie rozšíření se skládá z: použití spojení tabulek pomocí příkazu LEFT JOIN¹¹² a RIGHT JOIN, potvrzení (*Commit*) a zamítnutí (*Rollback*) transakce v průběhu provádění SQL příkazu v databázovém serveru a provedení stejného příkazu desetkrát, dvacetkrát, padesátkrát, stokrát a pětsetkrát v řadě za sebou.

¹¹⁰ Nejvyšší hranice datového typu INTEGER

¹¹¹ Firebird - SELECT CURRENT_DATE, CURRENT_TIMESTAMP FROM RDB\$DATABASE
Oracle – SELECT sysdate, systimestamp FROM DUAL

¹¹² JOIN = INNER JOIN zobrazí se pouze stejné záznamy z obou tabulek. LEFT JOIN = LEFT OUTER JOIN – zobrazí všechna data z levé tabulky a k nim připojí pravou tabulku. RIGHT JOIN = RIGHT OUTER JOIN – opak LEFT JOINU – analogicky zobrazí pravou tabulku

Použitá aplikace

Pro vlastní testování jsem díky vývojovému prostředí Eclipse¹¹³ a jazyku Java vytvořil jednoduchou konzolovou aplikaci¹¹⁴, která vytvoří spojení s databází, vygeneruje náhodné parametry pro uvedený SQL příkaz a zaznamená pouze dobu vlastního provádění v databázi. Pro propojení této aplikace a databázového systému Firebird jsem použil JDBC¹¹⁵ rozhraní Jaybird 2.1.6¹¹⁶ a pro propojení databáze Oracle pak Oracle Thin JDBC¹¹⁷. Vlastní testovací aplikaci jsem naprogramoval z toho důvodu, aby výsledná doba provádění SQL příkazu v databázi byla očištěna o dobu nutnou pro vytvoření spojení s databází (na začátku) a dobou nutnou pro zobrazení výsledků (na konci). Na konci aplikace je blok příkazů, který zaručí, aby se testované změny dat (např. vložení nových řádků) zamítnuly (*rollback*).

5.2. Postup testování

Po vytvoření tabulek, naplnění daty, následné implementaci testovací aplikace jsem definoval postup (metodiku), kterým se budu řídit při vlastním testování.

Na základě dlouhodobého pozorování výsledků provádění SQL příkazu v databázi, jsem odvodil, že stabilní výsledky u obou databázových systémů se začínají objevovat kolem padesátého opakování SQL příkazu.

Vlastní testování spočívalo ve spouštění SQL příkazů uvedených v příloze 2 v testovací aplikaci a zaznamenání výsledné doby provádění v milisekundách.

¹¹³ IDE Eclipse [online]. Dostupné na:
<http://www.eclipse.org/downloads/>

¹¹⁴ Zdrojové kódy k dispozici na:
http://www.edisk.cz/stahnout-soubor/65648/xlelr00-testovaci-aplikace.rar_6.9KB.html

¹¹⁵ JDBC = Java Database Connectivity – rozhraní pro propojení jazyka Java s databázovým serverem

¹¹⁶ JayBird 2.1.6 [online]. Dostupné na:
<http://www.firebirdsql.org/index.php?op=files&id=jaybird>

¹¹⁷ Oracle Thin JDBC [online]. Dostupné na:
http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/htdocs/jdbc_10201.html

Proti zabránění rychlejšího zpracování SQL příkazu z paměti, jsem číselné i znakové parametry podmínek generoval náhodně pomocí jazyka Java. (Například WHERE C1 = náhodné číslo AND C2 = náhodný řetězec znaků).

Pro zachování stejných podmínek bylo nutné v databázovém systému Oracle vytvořit sekvenci a spouštět pro generování hodnot primárních klíčů.

5.3. Dosažené výsledky

Tato část prezentuje výsledky obou databázových systémů získané v testovací aplikaci.

Pro lepší prezentaci a pochopení dosažených výsledků jsem určil statistické veličiny – minimum, maximum, modus, četnost, rozptyl a směrodatnou odchylku a střední hodnotu. Modus představuje dobu provádění SQL příkazu, která se při padesáti opakováních vyskytovala nejčastěji. Četnost znamená, kolikrát se doba provádění (modus) objevila v celém testovacím souboru. Navíc jsem ještě připočetl hodnoty o jedna větší nebo menší, které mohly vzniknout při převodu časových hodnot. Minimum (a maximum) představují nejnižší (a nejvyšší) dosaženou dobu trvání provádění SQL příkazu v databázovém serveru. Rozptyl je definován jako součet čtvercových odchylek od střední hodnoty. Představuje variabilitu testovacího souboru. Pro snazší představu o variabilitě jsem z rozptylu určil odmocninu – tzv. směrodatnou odchylku. Vyšší hodnota představuje větší různorodost výsledků. Nakonec jsem přidal střední hodnotu (medián), který výsledky rozděluje na dvě stejně početné kategorie.

Všechny detailní výsledky jsou uvedeny v příloze 3, zde prezentuji jen nejdůležitější tři veličiny – modus = vlastní doba provádění, četnost = počet „správných“ výsledků v padesáti opakováních a střední hodnotu, která představuje bod, kde polovina hodnot leží pod a druhá polovina výsledků nad tímto bodem. Nižší hodnoty u doby provádění znamenají vyšší rychlost provádění a tedy lepší výsledky.

Výsledky obou databázových serverů zobrazují následující tabulky 3, 4, 5

5.3.1. Výběry dat

Tabulka 3 – Dosažené výsledky databázových systémů v kategorii výběry dat

Kategorie příkazu	Doba trvání příkazu		Četnost		Střední hodnota		Oracle x Firebird
	Firebird	Oracle	Firebird	Oracle	Firebird	Oracle	
Výběr dat obsahující jednoduchou podmínku	94	31	26	27	92	32	Oracle
Výběr dat obsahující agregaci	78	31	36	38	83	31	Oracle
Výběr dat obsahující agregaci v podmínce	78	31	35	29	83	32	Oracle
Výběr dat a použití jednoho příkazu JOIN	187	47	27	36	194	47	Oracle
Výběr dat a použití více příkazů JOIN	78	78	34	10	83	102	Firebird = Oracle
Výběr dat a CROSS JOIN s více příkazy JOIN	94	94	32	45	89	94	Firebird = Oracle
Výběr dat obsahující vnořený výběr dat v podmínce	78	47	33	43	83	47	Oracle
Výběr dat obsahující vnořený výběr dat	188	47	25	33	196	47	Oracle
Spojení více výběrů dat pomocí příkazu UNION	391	125	30	23	399	110	Oracle

V kategorii „výběr dat a použití více příkazů JOIN“ a „výběr dat a CROSS JOIN s více příkazy JOIN“ oba databázové systémy prokázaly stejné doby trvání. Ve zbývajících kategoriích týkajících se výběrů dat vychází vítězně databázový systém Oracle Express edice.

5.3.2. Operace s daty (přidávání, aktualizace, mazání)

Tabulka 4 - Dosažené výsledky databázových systémů v kategorii operace s daty

Kategorie příkazu	Doba trvání příkazu		Četnost		Střední hodnota		Oracle x Firebird
	Firebird	Oracle	Firebird	Oracle	Firebird	Oracle	
Aktualizace sloupce pomocí příkazu UPDATE	141	16	27	36	144	16	Oracle
Aktualizace sloupce obsahující příkaz JOIN	208047	547	2	28	204571	531	Oracle
Vložení dat pomocí příkazu INSERT	62	16	43	35	65	16	Oracle
Vložení dat pomocí příkazu INSERT obsahující SELECT a JOIN	500	875	26	24	546	891	Firebird
Smazání dat použitím příkazu DELETE	109	172	44	39	112	187	Firebird
Smazání dat použitím příkazu DELETE obsahující agregaci	261312	85875	2	11	296158	85359,5	Oracle

V kategorii aktualizace sloupce obsahující příkaz JOIN databázový systém Firebird prokázal značný nárůst doby trvání až na přibližně 3,5 minut. Stejně tak v kategorii smazání dat použitím příkazu DELETE obsahující agregaci se doba trvání příkazu vyšplhala až na přibližně 4,5 minut. Na druhou stranu ale prokázal lepší výsledky při vkládání a mazání dat. Oracle Express edice také zaznamenala nárůst doby trvání u poslední kategorie, celkově až na 1,4 minut, ale i přesto byl rychlejší než databázový systém Firebird.

5.3.3. Rozšíření

Tabulka 5 - Dosažené výsledky databázových systémů v kategorii rozšíření

Kategorie příkazu	Doba trvání příkazu		Četnost		Střední hodnota		Oracle x Firebird
	Firebird	Oracle	Firebird	Oracle	Firebird	Oracle	
Odvolání změn (ROLLBACK)	422	125	9	8	433	172	Oracle
Potvrzení změn (COMMIT)	250	47	21	15	301	47	Oracle
Vnitřní spojení (LEFT JOIN)	78	47	28	38	90	47	Oracle
Vnější spojení (RIGHT JOIN)	78	47	36	38	82	47	Oracle
10 opakování	78	16	28	40	71	16	Oracle
20 opakování	78	31	35	26	83	31	Oracle
50 opakování	125	47	24	25	117	39	Oracle
100 opakování	172	63	39	38	169	63	Oracle
500 opakování	625	281	11	40	647	281	Oracle

Ve všech kategoriích z oblasti rozšíření s přehledem zvítězil databázový systém Oracle.

6. Shrnutí dosažených výsledků

V této části bych chtěl čtenáři poskytnout ucelené shrnutí všech dosažených výsledků porovnávání podle oblastí a kritérií definovaných na začátku této práce.

Databázový systém Firebird je zdarma distribuován pod licencí IPL, a to včetně zdrojových kódů. Oracle bychom mohli považovat z hlediska licencí za zkušební verzi placeného komerčního databázového systému (omezená verze je zdarma).

Protože se jedná o zkušební produkt, obsahuje Oracle Express Edition omezení z hlediska operačních systémů, hardwaru, uložení uživatelských dat i poskytnutých rozhraní. Tato čtyři kritéria mohou výrazně ovlivnit konečný výběr databázového systému.

Architekturu databázového serveru řeší oba systémy zcela odlišně: Firebird používá multigenerační architekturu založenou na vytváření více verzí záznamu a evidenci stavu transakcí. Neblokování zápisu při čtení dat zaručuje výhodu větší propustnost než systémy se zámkou a transakčním protokolem, který používá databázový systém Oracle Express edice.

Další rozdíl lze vypožorovat v systému ukládání dat na disku. V databázi Oracle se ukládají informace (uživatelská data, přístup k databázi apod.) do několika typů souborů (datové soubory, kontrolní soubory, soubory transakčního protokolu). Oproti tomu Firebird uchovává veškeré informace v jednom souboru.

V instalaci obou databázových systémů nejsou možnosti výběru pokročilejších funkcí, liší se tedy pouze nároky na volné místo disku.

Pro administraci obou databází jsou dodávány nástroje dostupné přes příkazový řádek. Větší uživatelskou přívětivost zajišťují dostupné administrační aplikace. Oracle navíc poskytuje webové administrační rozhraní.

Oba databázové systémy nabízí datové typy v rozsahu standardu SQL92 a dále ho rozšiřují. Firebird pomocí datových typů BIGINT a BLOB, Oracle pak přes NVARCHAR2, INTERVAL day to second, INTERVAL year to month, TIMESTAMP with time zone, TIMESTAMP with local time, LONG, RAW, LONG RAW, BLOB, CLOB, NCLOB a BFILE. Existuje také možnost definování vlastních datových typů pomocí domén (Firebird) a souborných datových typů (Oracle).

Databázový systém Firebird rovněž umožňuje vytvořit dotazovatelné procedury, definovat vlastní (uživatelské) výjimky i externí funkce. Oracle pro ukládání agregovaných výsledků používá materializované pohledy, lze definovat funkce, které jsou spolu s uloženými procedurami organizovány do balíčků, a navíc alternativní názvy objektů poskytují synonyma. Oba databázové systémy umožňují definovat spouště, liší se pouze úrovní, na které se použijí.

Zajímavou funkcionalitu představuje Fyrcle, který představuje přechod od komerčního softwaru k open source. Pro každou databázi Firebird lze vytvořit stín, který ji zastoupí v případě pádu serveru, na kterém je spuštěna stínovaná databáze a zachovat tak její dostupnost pro klienty. Oracle vsadil na vytváření jednoduchých webových aplikací pomocí Oracle Application Express. Není zde nutná znalost programování, vše je vytvořeno pomocí průvodců a konfiguračních obrazovek. Jako jednoduchá ukázka možností vyšších edic byla použita technologie Flashback (*Flashback Query*), která umožňuje vytvářet dotazy na změněná data k časovému okamžiku v minulosti.

Mezi hlavní výhody multigenerační architektury používané databázovým systémem Firebird patří skutečnost, že transakce¹¹⁸, která vyžaduje stabilní pohled na data (*Repeatable Read*), nezabraňuje ostatním transakcím změny dat (ani je nezpomaluje). Stejně tak nepotvrzené změny nikdy nebrání čtení záznamů jiným transakcím (jsou ignorovány). Multigenerační architektura nevyžaduje použití zámků (kromě izolační úrovně *Serializable*), protože případné konflikty při souběžné aktualizaci záznamů lze najít v seznamu verzí daného řádku. U systémů se zámků (Oracle) je zaručeno, že transakce může jednou přečtené řádky rovněž změnit (díky zámku proti zápisu nastavenému při čtení). Databázový systém Firebird ale kvůli absenci zámků nijak nezaručuje, že jednou přečtený řádek bude možné rovněž změnit. Pokud je taková funkcionalita vyžadována, je třeba použít vyšší izolační úroveň (*Serializable*). Výhodu transakčního protokolu představuje lehce realizovatelné přírůstkové zálohování. Nastavením izolační úrovně můžeme velkou mírou ovlivnit propustnost databáze.

¹¹⁸ Pojem „transakce“ znamená, že se jedná o uspořádané bloky příkazů, které mají zabezpečit konzistenci dat.

Pro zálohování a obnovu dat v základní instalaci oba databázové systémy poskytují utility dostupné přes příkazový řádek. Databáze Firebird lze na rozdíl od Oracle Express Edition zálohovat pouhým kopírováním souboru databáze nebo vygenerováním SQL skriptu obsahující data. Obě databáze lze zálohovat i za plného provozu.

Pomoc při jakýchkoliv potížích s databázovými servery lze nalézt ve velkém množství informačních zdrojů – část z nich je obsažena již při instalaci, zbývající je k dispozici na oficiálních stránkách výrobce nebo diskusních fórech.

Ze všech tří testovaných oblastí vyšla vítězně Oracle Express edice, jen ve dvou kategoriích (výběr dat a použití více příkazů JOIN“ a „výběr dat a CROSS JOIN s více příkazy JOIN) byl výkonnější databázový systém Firebird.

Databázový systém Oracle je vhodný použít zejména v menších podnicích, které osloví cenou i poskytnutými funkcemi. Oproti tomu Firebird je vhodný zejména pro jednouživatelské aplikace (např. pro mobilní zařízení), víceuživatelské aplikace pro operační systém Windows případně pro demoverze nebo katalogy pro disky CD-ROM.

Souhrnný pohled na výsledky porovnání podle kritérií zobrazuje tabulka 6.

Tabulka 6 - Výsledky porovnávání dle kritérií

Kritérium/databáze	Firebird	Oracle Express Edition
Operační systém	Žádné	Windows, Linux (32bit)
Hardware	Architektura Super-Server neumí využít více jader procesoru	max. 1 CPU, 1 GB RAM
Velikost databáze	neomezená	5GB (4GB + 1 GB)
Aplikační rozhraní	Delphi, C, C++, Java, PHP, Python, Perl i .NET, JDBC, ODBC a OLEDB	C, C++, .NET, ODBC,OLE DB a JDBC, PHP Nepodporuje Javu

Kritérium/databáze	Firebird	Oracle Express Edition
Architektura	Multigenerační architektura s použitím evidence stavu transakcí a více verzí záznamů	Fyzická, logická a paměťová

Kritérium/databáze	Firebird	Oracle Express Edition
Instalace	Instalační balík 6,5 MB	Instalační balík 157 MB
	Instalace 20,5 MB	Instalace 1GB

Kritérium/databáze	Firebird	Oracle Express Edition
Administrace	SQL Manager Lite 2008	SQL developer
	ISQL (příkazový řádek)	vestavěné webové administrační rozhraní
		SQL*Plus(příkazový řádek)

Kritérium/databáze	Firebird	Oracle Express Edition
Navíc oproti standardu SQL92	BIGINT, BLOB	NVARCHAR2, INTERVAL day to second, INTERVAL year to month, TIMESTAMP with time zone, TIMESTAMP with local time, LONG, RAW, LONG RAW, BLOB, CLOB, NCLOB,BFILE
Definice vlastních datových typů	Domény	Souborné typy

Kritérium/databáze	Firebird	Oracle Express Edition
Databázové objekty	Standardní a dotazovatelné uložené procedury	Materializované pohledy
	Uživatelsky definované výjimky	Funkce
	Uživatelsky definované funkce	Balíčky
	Spouště - akce na úrovni databáze, tabulky, transakce	Spouště - akce na úrovni databáze, tabulky, transakce, systémové a uživatelské akce
		Synonyma

Kritérium/databáze	Firebird	Oracle Express Edition
Zvláštní funkcionality	Fyrcle - Oracle mód Firebirdu	Oracle Application Express - tvorba jednoduchých webových aplikací nad vlastní databází.
	Stínování databáze (Shadowing) – reálná kopie aktuální databáze pro případ pádu serveru	Oracle Flashback technologie - dotazy na historická data

Kritérium/databáze	Firebird	Oracle Express Edition
Úroveň izolace transakcí	Read Committed, Snapshot, Snapshot table stability	Read Committed, Serializable, Read-Only
Konzistence	multigenerační architektura	zámky, Multi-Version Read Consistency (zápis dat neblokuje jejich čtení)

Kritérium/databáze	Firebird	Oracle Express Edition
Řízení zálohy a obnovy dat	Utility GBAK, NBACKUP Aplikace Firebird Database Backup/Restore kopírování souboru databáze vygenerování SQL skriptu	Utilita RMAN (Recovery Manager)

Kritérium/databáze	Firebird	Oracle Express Edition
Technická podpora a dokumentace	Oficiální stránky projektu Firebird, portál FAQ, Release Notes, Quick Sheets, Google Groups, Yahoo Groups, komunita SourceForge.net	Fórum Oracle Express Edition, Oracle Documen- tation Library, portál Ask Tom Kyte

7. Závěr

V této bakalářské práci jsem porovnával open source databázový systém Firebird a (komerční) databázový systém Oracle. Cílem bylo porovnat je z hlediska uživatele, který se rozhoduje, zda systém, který je nabízen na trhu databázových systémů zdarma (open source), je výhodnější, zda je schopen uspokojit jeho požadavky na určité vlastnosti nebo výkon.

V první části této práce jsem definoval oblasti a kritéria porovnávání, na jejichž základě jsem oba databázové systémy srovnával. Z této části vyplynulo, že oba databázové systémy se liší především licencí, pod kterou jsou dodávány, omezeními, architekturou, unikátní funkcionalitou, řešením konzistence i výkonem.

Firebird je open source databázový systém, který je dodáván zdarma včetně zdrojových kódů. Oracle Express edice je jako jediná verze od společnosti Oracle ke stažení i komerčnímu nasazení zdarma. Oracle se tak snaží přilákat nové zákazníky - především menší firmy, které nemají potřebný rozpočet na zakoupení komplexního řešení. Předpokládá však, že s růstem společnosti, a tudíž i s růstem uživatelských dat pak přejdou na vyšší (placené) verze. Její největší nevýhodou jsou následující omezení: Oracle Express edici lze nainstalovat a provozovat pouze na 32 bitových verzích platform Windows a Linux, lze uložit pouze 4GB uživatelských dat, vždy bude využito pouze jedno procesorové jádro a 1GB paměti RAM a neposkytuje rozhraní pro jazyk Java.

Firebird používá multigenerační architekturu založenou na vytváření více verzí záznamu a evidenci stavu transakcí. Tato výhoda se nejvíce uplatní v aplikacích, ve kterých je vyžadováno komplexní zpracování dat souběžně s jejich aktualizací (např. on-line sběr dat s jejich současným vyhodnocováním). Mezi další výhody patří spolehlivost a okamžité zotavení po pádu systému - všechny aktivní transakce jsou označeny jako odvolané a při nejbližší příležitosti jsou odstraněny. V případě použití databázového systému Firebird, ale musíme vzít v úvahu i jeho nevýhody. Výkon databáze může výrazně ovlivnit zejména režie nutná ke zpracování více verzí řádků a režie pro odstranění nepotřebných verzí.

Dále záleží na typu architektury databázového systému Firebird. Architekturu Classic použijeme v aplikacích, ve kterých k databázi přistupuje jeden uživatel (například aplikace pro mobilní zařízení), variantu SuperServer lze bez obtíží použít v aplikacích, do kterých přistupuje více uživatelů, určených především pro operační systém Windows.

I přes obsažená omezení je výhodné použít Oracle Express edici jako testovací prostředí, které lze samostatně nainstalovat každému uživateli ve společnosti, která používá vyšší edici databáze Oracle. Další využití představuje nasazení Oracle Express edice jako hlavní databáze v malém projektu (například vytvoření vlastní Wikipedie). Open source databáze Firebird představuje alternativu, ke které se může přiklonit vedení malé firmy, pokud jejich nároky překračují limity Oracle Express edice a není dostatek peněz pro zakoupení vyšší edice.

Firebird představuje přechod od Oracle k Firebirdu. Je zde určitá snaha přilákat nové uživatele, kteří díky otevřenosti kódu mohou přispět svými znalostmi. Pro zachování vysoké dostupnosti databáze je k dispozici možnost tzv. stínování – vytvoření reálné kopie na jiném disku. Společnost Oracle se dala cestou rozšiřování možnosti práce s Oracle Express edicí a poskytuje tedy tvorbu jednoduchých webových aplikací pomocí Oracle Application Express.

Jsem přesvědčen, že cíle porovnat databázový systém Firebird a Oracle a poskytnout o nich vyčerpávající informace byly splněny v plném rozsahu.

V následující části jsem prakticky ověřil výkon obou databázových systémů. Z analýzy informačních zdrojů jsem nabyl dojmu, že databázový server Firebird bude rychlejší. Ale tento předpoklad byl praktickým ověřením výkonu vyvrácen. Ve většině kategorií (kromě kategorií vložení dat pomocí příkazu INSERT obsahující SELECT a JOIN a smazání dat použitím příkazu DELETE) se jako výkonnější, a tedy rychlejší z obou srovnávaných databází jeví Oracle Express edice.

Z toho je patrné, že databázový systém Firebird, ačkoliv má řadu nesporných výhod, je neomezený a poskytuje více operačních systémů, nemůže v oblasti výkonu plně konkurovat komerčnímu řešení dodávanému společností Oracle.

Musím konstatovat, že cíl prakticky ověřit výkon obou jmenovaných databázových systémů nemohl být splněn do všech detailů, neboť praktické porovnání výkonu by mohlo probíhat i za jiných podmínek.

Mohla být třeba zvolena jiná architektura nebo by databáze mohla být nasazena na serveru.

Tento cíl nebyl úplně splněn vzhledem k omezenému rozsahu bakalářské práce. Ze stejného důvodu nebylo možné ve všech oblastech porovnávání dojít až do detailů.

Velmi zajímavé by mohlo být detailní porovnání mechanismů multigenerační architektury s fungováním transakčního protokolu nebo přenositelnost SQL kódu uložených procedur a spouští mezi databázovými systémy. Do této práce jsem se snažil vybrat nejdůležitější funkce nebo jedinečné vlastnosti z obou databází. Možnosti rozšíření této práce spatřuji především v prozkoumání oblasti ladění a optimalizaci výkonu, exportu a importu dat.

Myslím, že moje bakalářská práce splnila vytčené cíle v rozhodujícím rozsahu a i přes některé nedostatky je přínosná svým jedinečným porovnáním dvou rozdílných databázových systémů v teoretické rovině a v praktickém otestování jejich výkonu, dále pak tím, že poskytuje řadu užitečných informací pro uživatele databázových systémů i některé podněty pro další teoretické zkoumání a testování v této oblasti.

8. Přílohy

8.1. Příloha 1 - Aktuální ceník licencí Oracle

Section I

Prices in USA (Dollar)

Database Products	Oracle Database			
	Named User Plus	Software Update License & Support	Processor License	Software Update License & Support
Oracle Database				
Standard Edition One	180	39.60	5,800	1,276.00
Standard Edition	350	77.00	17,500	3,850.00
Enterprise Edition	950	209.00	47,500	10,450.00
Personal Edition	460	101.20	-	-
Lite Mobile Server	-	-	23,000	5,060.00
Lite Client	60	13.20	-	-
<i>Enterprise Edition Options:</i>				
Real Application Clusters	460	101.20	23,000	5,060.00
Real Application Clusters One Node	200	44.00	10,000	2,200.00
Active Data Guard	200	44.00	10,000	2,200.00
Partitioning	230	50.60	11,500	2,530.00
Real Application Testing	230	50.60	11,500	2,530.00
Advanced Compression	230	50.60	11,500	2,530.00
Total Recall	120	26.40	5,800	1,276.00
Advanced Security	230	50.60	11,500	2,530.00
Label Security	230	50.60	11,500	2,530.00
Database Vault	460	101.20	23,000	5,060.00
OLAP	460	101.20	23,000	5,060.00
Data Mining	460	101.20	23,000	5,060.00
Data Profiling and Quality	350	77.00	17,500	3,850.00
Spatial	350	77.00	17,500	3,850.00
In-Memory Database Cache	-	-	41,500	9,130.00
Retail Data Model	800	176.00	40,000	8,800.00

8.2. Příloha 2 – Použité SQL příkazy

8.2.1. Výběry dat

Výběr dat obsahující jednoduchou podmínku

```
SELECT *  
FROM TABLE_T01  
WHERE C7 > 123454321
```

Výběr dat obsahující agregaci

```
SELECT AVG(C7)  
FROM TABLE_T01  
WHERE C3 LIKE '%a%'
```

Výběr dat obsahující agregaci v podmínce

```
SELECT C1, C16, C3, C18, C25, C14  
FROM TABLE_T01  
WHERE C8 > 123454321  
      AND C4 LIKE '%a%'  
GROUP BY C1, C16, C3, C18, C25, C14  
HAVING AVG(C21)> 123454321
```

Výběr dat s použitím jednoho příkazu JOIN

```
SELECT *  
FROM TABLE_T01 T01  
      JOIN TABLE_T02 T02 ON T01.C1 = T02.C1
```

Výběr dat s použitím více příkazů JOIN

```
SELECT T01.C10, T01.C12, T01.C16, T02.C10, T03.C14,  
T04.C24, T04.C8  
FROM TABLE_T01 T01  
      JOIN TABLE_T02 T02 ON T01.PK = T02.PK  
      JOIN TABLE_T03 T03 ON T01.C1 = T03.C13  
      JOIN TABLE_T04 T04 ON T01.C1 = T04.C23  
WHERE T01.C14 > 123454321  
GROUP BY T01.C10, T01.C12, T01.C16, T02.C10, T03.C14,  
T04.C24, T04.C8  
HAVING T04.C8 > nahodneCislo2
```

Výběr dat s příkazem CROSS JOIN a použitím více příkazů JOIN

```
SELECT * FROM TABLE_T01 T01
CROSS JOIN TABLE_T02 T02
      JOIN TABLE_T03 T03 ON T01.PK = T03.PK
      JOIN TABLE_T04 T04 ON T01.C1 = T04.C5
```

Výběr dat obsahující vnořený výběr dat v podmínce

```
SELECT *
FROM TABLE_T05 T05
WHERE T05.C7 IN (SELECT C10 FROM TABLE_T07)
```

Výběr dat obsahující vnořený výběr dat

```
SELECT * FROM
      (SELECT T01.C10, T01.C24, T01.C17, T02.C6,
      T03.C13, T04.C11, T04.C3
      FROM TABLE_T01 T01
      JOIN TABLE_T02 T02 ON T01.PK = T02.PK
      JOIN TABLE_T03 T03 ON T01.C1 = T03.C13
      JOIN TABLE_T04 T04 ON T01.C1 = T04.C23)
WHERE C17 LIKE '%a%'
```

Spojení tří výběrů dat pomocí příkazu UNION

```
      SELECT C24, C20, C21 FROM TABLE_T01 T01
UNION
      SELECT T05.C4, T06.C17, T05.C13
      FROM TABLE_T05 T05
      JOIN TABLE_T06 T06 ON T05.C13 = T06.C11
UNION
      SELECT T09.C21, T09.C15, T08.C4
      FROM TABLE_T08 T08
      JOIN TABLE_T09 T09 ON T09.PK = T08.PK
UNION
      SELECT C13, C4, C20 FROM TABLE_T10 T10
```

8.2.2. Operace s daty (aktualizace, vložení, smazání)

Aktualizace sloupce

```
UPDATE TABLE_T07
SET C13 = 123454321
WHERE C10 LIKE '%a%'
```

Aktualizace sloupce obsahující JOIN

```
UPDATE TABLE_T07
SET C13 = (SELECT AVG (T10.C19)
FROM TABLE_T10 T10
JOIN TABLE_T02 T2 ON T2.PK = T10.PK
AND T10.C18 LIKE '%a%')
```

Vložení dat pomocí příkazu INSERT

```
INSERT INTO TABLE_T07 (C1, C2, C3, C4, C5, C6, C7, C8, C9,
C10, C11, C12, C13, C14, C15, C16, C17, C18) VALUES
(sysdatetime, sysdate, 11111, sysdatetime, sysdate,
sysdate, sysdate, sysdate, 'a', 'a', 22222, 222222, 39394,
'a', 23949, 'a', 12493, sysdatetime)
```

Vložení dat pomocí příkazu INSERT obsahující příkazy SELECT a JOIN

```
INSERT INTO TABLE_T07 (C1, C2, C3, C4, C5, C6, C7, C8, C9,
C10, C11, C12, C13, C14, C15, C16, C17, C18)

SELECT T09.C2 AS C1, T08.C2 AS C2, T09.C15 AS C3, T09.C21
AS C4, T09.C5 AS C5, T09.C11 AS C6, T08.C11 AS C7, T08.C13
AS C8, T08.C9 AS C9, T09.C9 AS C10, T08.C14 AS C11, T09.C8
AS C12, T08.C10 AS C13, T09.C18 AS C14, T08.C16 AS C15,
T08.C18 AS C16, T09.C17 AS C17, T09.C13 AS C18
FROM TABLE_T08 T08
JOIN TABLE_T09 T09 ON T09.PK = T08.PK
```

Smazání dat použitím příkazu DELETE

```
DELETE FROM TABLE_T09 WHERE C6 LIKE '%a%'
```

Smazání dat použitím příkazu DELETE obsahující agregaci

```
DELETE FROM TABLE_T09 WHERE C4 NOT IN (SELECT MIN (C4)
FROM TABLE_T09 GROUP BY C4)
```

8.2.3. Rozšíření

Použití rollback /commit

```
UPDATE TABLE_T07
SET C1 = CURRENT_TIMESTAMP
WHERE C10 LIKE '%a%'
```

Spojení dvou tabulek pomocí LEFT JOIN

```
SELECT * FROM TABLE_T01 T01  
LEFT JOIN TABLE_T02 T02  
ON T01.C1 T02.C1
```

Spojení dvou tabulek pomocí RIGHT JOIN

```
SELECT * FROM TABLE_T01 T01  
RIGHT JOIN TABLE_T02 T02  
ON T01.C1 T02.C1
```

Opakování příkazu

```
INSERT INTO TABLE_T07 (C1, C2, C3, C4, C5, C6, C7, C8, C9,  
C10, C11, C12, C13, C14, C15, C16, C17, C18) VALUES  
(sysdatetime, sysdate, 11111, sysdatetime, sysdate,  
sysdate, sysdate, sysdate, 'a', 'a', 22222, 222222, 39394,  
'a', 23949, 'a', 12493, sysdatetime)
```

8.3. Příloha 3 – Detailní rozbor výsledků

8.3.1. Výběry dat

Firebird

Tabulka 7 - Výsledky dosažené databázovým systémem Firebird – výběry dat

Kategorie příkazu	Doba provádění	Četnost	Minimum	Maximum	Rozptyl	Směrodatná odchylka	Střední hodnota
Výběr dat obsahující jednoduchou podmínku	94	26	63	172	242,05	15,56	92
Výběr dat obsahující agregaci	78	36	78	109	60,79	7,80	83
Výběr dat obsahující agregaci v podmínce	78	35	78	94	51,44	7,17	83
Výběr dat a použití jednoho příkazu JOIN	187	27	156	234	156,84	12,52	194
Výběr dat a použití více příkazů JOIN	78	34	78	94	51,27	7,16	83
Výběr dat a CROSS JOIN s více příkazy JOIN	94	32	78	110	63,46	7,97	89
Výběr dat obsahující vnořený výběr dat v podmínce	78	33	78	94	54,84	7,41	83
Výběr dat obsahující vnořený výběr dat	188	25	171	281	858,67	29,30	196
Spojení více výběrů dat pomocí příkazu UNION	391	30	390	438	139,39	11,81	399

Oracle

Tabulka 8 – Výsledky dosažené databázovým systémem Oracle – výběry dat

Kategorie příkazu	Doba provádění	Četnost	Minimum	Maximum	Rozptyl	Směrodatná odchylka	Střední hodnota
Výběr dat obsahující jednoduchou podmínku	31	27	31	63	71,74	8,47	32
Výběr dat obsahující agregaci	31	38	31	47	45,32	6,73	31
Výběr dat obsahující agregaci v podmínce	31	29	31	62	69,29	8,32	32
Výběr dat a použití jednoho příkazu JOIN	47	36	46	63	49,98	7,07	47
Výběr dat a použití více příkazů JOIN	78	10	62	297	3 901,78	62,46	102
Výběr dat a CROSS JOIN s více příkazy JOIN	94	45	78	94	22,35	4,73	94
Výběr dat obsahující vnořený výběr dat v podmínce	47	43	31	47	29,12	5,40	47
Výběr dat obsahující vnořený výběr dat	47	33	31	63	78,00	8,83	47
Spojení více výběrů dat pomocí příkazu UNION	125	23	109	141	69,96	8,36	110

8.3.2. Operace s daty (aktualizace, vložení, smazání)

Firebird

Tabulka 9 - Výsledky dosažené databázovým systémem Firebird – operace s daty

Kategorie příkazu	Doba provádění	Četnost	Minimum	Maximum	Rozptyl	Směrodatná odchylka	Střední hodnota
Aktualizace sloupce pomocí příkazu UPDATE	141	27	125	313	736,74	27,14	144
Aktualizace sloupce obsahující příkaz JOIN	208 047	2	157 188	230 703	204 628 315,76	14 304,84	204 571
Vložení dat pomocí příkazu INSERT	62	43	62	93	43,52	6,60	65
Vložení dat pomocí příkazu INSERT obsahující SELECT a JOIN	500	26	484	1 328	17 935,66	133,92	546
Smazání dat použitím příkazu DELETE	109	44	109	141	39,93	6,32	112
Smazání dat použitím příkazu DELETE obsahující agregaci	261 312	2	260 859	510 125	2 214 702 978,76	47 060,63	296 158

Oracle

Tabulka 10 - Výsledky dosažené databázovým systémem Oracle – operace s daty

Kategorie příkazu	Doba provádění	Četnost	Minimum	Maximum	Rozptyl	Směrodatná odchylka	Střední hodnota
Aktualizace sloupce pomocí příkazu UPDATE	16	38	15	32	44,05	6,64	16
Aktualizace sloupce obsahující příkaz JOIN	547	8	203	640	8 747,24	93,53	531
Vložení dat pomocí příkazu INSERT	16	41	15	78	151,97	12,33	16
Vložení dat pomocí příkazu INSERT obsahující SELECT a JOIN	875	6	828	4 515	383 385,22	619,18	891
Smazání dat použitím příkazu DELETE	172	13	125	296	1 481,61	38,49	187
Smazání dat použitím příkazu DELETE obsahující agregaci	85 875	2	84 578	86 813	266 149,01	515,90	85 360

8.3.3. Rozšíření

Firebird

Tabulka 11 - Výsledky dosažené databázovým systémem Firebird – rozšíření

Kategorie příkazu	Doba provádění	Četnost	Minimum	Maximum	Rozptyl	Směrodatná odchylka	Střední hodnota
Odvolání změn (ROLLBACK)	422	9	390	531	1 121,33	33	433
Potvrzení změn (COMMIT)	250	21	301	218	9 796,44	99	250
Vnitřní spojení (LEFT JOIN)	78	28	78	156	300,11	17	90
Vnější spojení (RIGHT JOIN)	78	36	78	94	48,01	7	82
10 opakování	78	28	62	79	60,76	8	71
20 opakování	78	35	78	94	51,79	7	83
50 opakování	125	24	109	141	71,61	8	117
100 opakování	172	39	156	204	64,36	8	169
500 opakování	625	11	609	734	936,60	31	647

Oracle

Tabulka 12 - Výsledky dosažené databázovým systémem Oracle – rozšíření

Kategorie příkazu	Doba provádění	Četnost	Minimum	Maximum	Rozptyl	Směrodatná odchylka	Střední hodnota
Odvolání změn (ROLLBACK)	125	8	125	484	5 967,93	77,25	172
Potvrzení změn (COMMIT)	47	15	31	344	2 407,03	49,06	47
Vnitřní spojení (LEFT JOIN)	47	38	46	63	44,08	6,64	47
Vnější spojení (RIGHT JOIN)	47	38	46	63	44,55	6,67	47
10 opakování	16	40	1	16	35,07	5,92	16
20 opakování	31	26	15	32	60,04	7,75	31
50 opakování	47	25	31	47	61,30	7,83	39
100 opakování	63	38	62	79	44,69	6,69	63
500 opakování	281	40	265	297	48,89	6,99	281

9. Zdroje

9.1. Seznam použité literatury

TIŠTĚNÉ ZDROJE

Buchalcevořá, A., Stanovská, I., Šimunek, M.: Základy softwarového inženýrství – základní témata, 1.vyd., skripta VŠE, 2002, ISBN 80-245-0346-8

Kevin Loney, Marlene Theriault: Mistrovství v Oracle – Kompletní průvodce tvorbou, správou a údržbou databází, 1. vydání, nakladatelství Computer Press, 2002, ISBN 80-7226-635-7

Pavel Císař: InterBase/Firebird – Tvorba, administrace a programování databází, 1. vydání, nakladatelství Computer Press, 2003, ISBN 80-7226-946-1

ELEKTRONICKÉ ZDROJE

Ask Tom Kyte [online]. Dostupné na:

<http://asktom.oracle.com/pls/asktom/f?p=100:1:0> 50

Classic and Superserver architectures [online]. Dostupné na:

<http://firebirdsql.org/manual/ufb-about-arch.html> 27

Client and server combined: Firebird Embedded Server [online]. Dostupné na:

<http://www.firebirdsql.org/manual/fbmetasecur-embedded.html> 18

Client and server combined: Firebird Embedded Server [online]. Dostupné na:

<http://www.firebirdsql.org/manual/ufb-cs-embedded.html> 18

Datagenerator 0.35[online]. Dostupné na:

<http://sourceforge.net/projects/datagenerator/> 51

Edice Oracle Database 11g [online]. Dostupné na:

<http://www.oracle.com/global/cz/database/edice.html> 12

EMS SQL Manager for InterBase/Firebird [online]. Dostupné na:

<http://sqlmanager.net/en/products/ibfb/manager> 29

FAQ – What are the technical limits of Firebird? [online] Dostupné na:

<http://www.firebirdfaq.org/faq61/> 14

FAQ - What's the limit on the Firebird database size? [online] Dostupné na:

<http://www.firebirdfaq.org/faq59> 14

Firebird – Overview of Features [online]. Dostupné na:

<http://firebirdsql.org/manual/ufb-about-features.html> 14

Firebird – System Requirments [online]. Dostupné na:

<http://firebirdsql.org/manual/ufb-about-sysreq.html> 14

Firebird Database Backup/Restore V1.0 [online]

http://download.cnet.com/FireBird-Backup-Restore/3000-2242_4-10688595.html 46

Firebird Documentation Index [online]. Dostupné na:	
http://firebirdsql.org/index.php?op=doc	48
Firebird Embedded a .NET – Jiří Činčura – Vyvojari.cz [online]. Dostupné na:	
http://www.vyvojar.cz/Articles/446-firebird-embedded-a-net.aspx	18
Firebird Foundation [online]. Dostupné na:	
http://www.firebirdsql.org/index.php?op=ffoundation	6
Forum: Oracle Database 10g Express Edition [online] Dostupné na:	
http://forums.oracle.com/forums/forum.jspa?forumID=251	49
Fyracle: Oracle-mode Firebird – [online]. Dostupné na:	
http://www.janus-software.com/fb_fyracle.html	38
Fyracle: Oracle-mode Firebird – [online]. Dostupné na:	
http://www.janus-software.com/fyracle.pdf	38
Google Groups – Firebird (CZ) [online]. Dostupné na:	
http://groups.google.com/group/firebird_cz	49
IBPhoenix Documentation [online]. Dostupné na:	
http://www.ibphoenix.com/main.nfs?a=ibphoenix&page=ibp_document	49
IDE Eclipse [online]. Dostupné na:	
http://www.eclipse.org/downloads/	53
Initial Developer's Public Licence Version 1.0 [online]. Dostupné na:	
http://www.firebirdsql.org/index.php?op=doc&id=idpl	11
InterBase Public Licence [online]. Dostupné na:	
http://www.firebirdsql.org/index.php?op=doc&id=ipl	11
IPL - InterBase Public License [online] Dostupné na:	
http://www.firebirdsql.org/index.php?op=doc&id=ipl	6
JayBird 2.1.6 [online] Dostupné na:	
http://www.firebirdsql.org/index.php?op=files&id=jaybird	53
Mozilla Public Licence [online]. Dostupné na:	
http://www.mozilla.org/MPL/MPL-1.1.html	11
Oracle Architectural Components [online]. Dostupné na:	
http://www.oracle.com/technology/tech/migration/isv/docs/OracleArchitectureOverview.pdf	24
Oracle Database 10g Express Edition Downloads[online]. Dostupné na:	
http://www.oracle.com/technology/software/products/database/xe/index.html	12
Oracle Database Concepts 10G - Data Concurrency and Consistency [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/consist.htm#sthref1944	45
Oracle Database Concepts 10G - Datafiles and Tablespace [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/img/cncpt037.gif	21

Oracle Database Concepts 10G - Introduction to the Oracle Database [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/intro.htm#i60798	21, 24
Oracle Database Concepts 10G - Oracle Memory Architecture [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/img/cncpt151.gif	25
Oracle Database Concepts 10G - Oracle Memory Architecture [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/memory.htm#g30926	23, 24
Oracle Database Concepts 10G - Oracle Process Architecture[online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/process.htm#i16977	25
Oracle Database Concepts 10G - Schema Objects [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/schema.htm#i20690	36
Oracle Database Concepts 10G - SQL, PL/SQL, and Java [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/sqlplsql.htm#sthref3652	37
Oracle Database Concepts 10g - Tablespaces, Datafiles, and Control Files [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/physical.htm#i2006	20
Oracle Database Concepts 10G - Triggers [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/triggers.htm#i10900	37
Oracle Database Concepts 10G -Object Datatypes and Object Views [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B19306_01/server.102/b14220/objects.htm#g446482	32
Oracle Database Express Edition 2 Day Developer Guide 10g [online]. Dostupné na:	
http://download.oracle.com/docs/cd/B25329_01/doc/appdev.102/b25108/xedev_intro.htm	15
Oracle Database Lite 10g [online].	
Dostupné na: http://www.oracle.com/database/lite_edition.html	13
Oracle Documentation Library [online]. Dostupné na:	
http://www.oracle.com/pls/xe102/homepage	49
Oracle Flashback Technology [online]. Dostupné na:	
http://www.oracle.com/technology/deploy/availability/htdocs/Flashback_Overview.htm	39
Oracle SQL Developer 1.5 – [online]. Dostupné na:	
http://www.oracle.com/technology/software/products/sql/index.html	29
Oracle Thin JDBC [online]. Dostupné na:	
http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/htdocs/jdbc_10201.html	53
Oracle's 30th Anniversary (PDF) [online] Dostupné na:	
http://www.oracle.com/oramag/profit/07-may/p27anniv_timeline.pdf	7
Release Notes Firebird 2.1 [online]. Dostupné na:	
http://firebirdsql.org/rlsnotes/rlsnotes210.html#rnf210-dbtriggers	34
Sourceforge.net [online]. Dostupné na:	
http://sourceforge.net/support/getsupport.php?group_id=9028	49

Standard SQL 92 [online] Dostupné na:	
http://www.contrib.andrew.cmu.edu/~shadow/sql/sql1992.txt	5
The Firebird FAQ [online]. Dostupné na:	
http://www.firebirdfaq.org/	48
The InterBase SuperServer Architecture [online]. Dostupné na:	
http://www.ibphoenix.com/main.nfs?a=ibphoenix&page=ibp_ss_vs_classic	18
The Relationships Among Segments, Extents, and Data Blocks [online]. Dostupné na:	
http://download-uk.oracle.com/docs/cd/B19306_01/server.102/b14220/img/cncpt027.gif	22
W3school – SQL Syntax [online] Dostupné na:	
http://www.w3schools.com/sql/sql_syntax.asp	5
Yahoo1 Groups - Support for Users of Firebird Releases [online]. Dostupné na:	
http://tech.groups.yahoo.com/group/firebird-support/messages	49

9.2. Seznam použitých obrázků

Obrázek 1 - Tabulkový prostor složený ze dvou datových souborů	21
Obrázek 2 - Vztah datových bloků, rozsahů a segmentů	22
Obrázek 3 – Komponenty systémové globální paměti	24
Obrázek 4 – Programová globální oblast a systémová globální oblast	25

9.3. Seznam tabulek

Tabulka 1– Průběh transakce	17
Tabulka 2– Srovnání typů architektur Classic a SuperServer	27
Tabulka 3 – Dosažené výsledky databázových systémů v kategorii výběry dat	55
Tabulka 4 – Dosažené výsledky databázových systémů v kategorii operace s daty	56
Tabulka 5 – Dosažené výsledky databázových systémů v kategorii rozšíření	57
Tabulka 6 – Výsledky porovnávání dle kritérií	60
Tabulka 7 – Výsledky dosažené databázovým systémem Firebird – výběry dat	72
Tabulka 8 – Výsledky dosažené databázovým systémem Oracle – výběry dat	73
Tabulka 9 – Výsledky dosažené databázovým systémem Firebird – operace s daty	74
Tabulka 10 – Výsledky dosažené databázovým systémem Oracle – operace s daty	75
Tabulka 11 – Výsledky dosažené databázovým systémem Firebird – rozšíření	76
Tabulka 12 – Výsledky dosažené databázovým systémem Oracle – rozšíření	77