Velké databáze
High Performance Databases

Jan Géryk,
IS MU

Služby počítačových sítí, 21. 11. 2012

Témata přednášky

1. Databáze, historie
2. Databáze jako služba počítačové sítě
3. Vlastnosti DB systémů
4. Zpracování transakcí
5. Architektury rozsáhlých databází
6. Distribuované databázové systémy

Databáze, historie

- Co je databáze?
  - uspořádaná množina dat uložená na paměťovém médiu
  - nástroje pro manipulaci, správu a přístup
- Historie
  - 50. léta, COBOL
  - architektura síťových a hierarchických db
  - 70. léta, relační db, SQL
  - 80. léta, zlatý věk DB
  - objektové db, kombinace
- Relační architektura je dnes nejrozšířenější

Relační databáze

- založené na relačním modelu
- souvisí s teorií množin
  - realizuje podmnožinu kartézského součinu
- relace jsou reprezentovány tabulkami
  - dvourozměrné struktury
  - řádky chápeme jako záznamy
  - sloupce uchovávají info o relacích (atributy)
  - atributy mají svůj datový typ a doménu
- všechny základní operace pomocí jednoduchých funkcí
  - sjednocení, rozdíl, selekce, ...
  - relační kalkul a algebra

Databáze jako služba sítě

- DB jako mechanismus přístupu k datům

  - jednotné rozhraní pro efektivní vývoj a provoz malých aplikací

  - efektivní vzhledem ke konkrétním typům dat

  - využívá se služeb operačního systému

  - pouze jednouživatelský režim (více uživatelů => síťové FS)

  - nemožnost transakčního zpracování

Databáze jako služba sítě 2

- DB jako služba sítě

  - služba obsluhující aplikace přes vlastní jednotné rozhraní po síti

  - implementace vlastních síťových protokolů, ne sdílený síťový FS

  - server <-> více klientů = striktně klient – server pohled

  (klientem z hlediska databáze může být i aplikační server)

  - služba odstíněná od uživatelů dalšími vrstvami

  (aplikační, prezentační vrstvy)

Vlastnosti DB systémů

- Transakční systémy (banky, e-shopy)
- Transakce (převod peněz)
  - skupina příkazů, které převedou DB z jednoho konzistentního stavu do druhého
  - musí být provedeny všechny nebo žadný
- Dotazovací jazyk pro práci s daty v relační DB
  - SQL
  - původně navržen pro koncové uživatele (manažery)
  - syntakticky blizký angličtině
- SQL:
  - manipulace s daty: insert, update, delete
  - řízení transakcí: commit, rollback

Vlastnosti DB systémů 2

Systém pro zpracování transakcí splňující:

- Atomičnost
  - Transakce jsou zpracovány jako celek
- Konzistence
  - transakce uchovávají databázi v konzistentním stavu
- Izolovanost
  - jednotlivé operace jsou prováděny izolovaně vůči ostatním trans.
  - nesmím vidět změněná data jiné transakce
- Trvanlivost
  - data úspěšně ukončených transakcí musí být uložena trvanlivě, i po výpadku napájení

Zpracování transakcí

- Realizace – zámky
  - základní nástroje transakčního zpracování
  - ochrana proti nechtěným změnám
  - zamykání na úrovni řádků (ne tabulek, bloků), dnes už standard
  - vše zajišťuje databázový systém
- Kompromis: propustnost x konzistence
  - „read commited“ režim
    - každý příkaz čte jen to co bylo commitnuto
    - ochrana před čtením dočasných dat (verze dat)
    - během transakce může dojít ke změně hodnoty

Zpracování transakcí 2

- Deadlock
  - vzájemné zablokování
  - prevence: predikce, **detekce:** ukončí jednu z nich, obcházení: nejdřív uzamče vše, neřešení: speciální případy
  - kultura programování
    - pořadí zámků
    - co nejkratší dobu
- Oracle uvolní všechny zámky po skončení transakce

Architektury rozsáhlých DB

- Principy spojení klienta s DB
  - aplikační rozhraní (Aplikace v Javě)
  - klientské knihovny (Java - JDBC)
  - spojení, session (Oracle NET)
- Způsoby zpracování požadavku (server)
  - Oracle NET: vrstva zajišťující komunikaci mezi serverem a klientem, beží na obou
  - proces listener (port 1521)
  - vyhrazený server (dedicated): vždy nový proces, batch
  - sdílený server (shared)
    - výrazně snižuje nároky na paměť
    - lepší škálovatelnost, více současných spojení

Vlastnost: dostupnost

- Dnešní doba vyžaduje 24/7
- Ochrana před chybou (uživatel, HW, SW)
  - Uživatel: transakce (rollback), flashback (i DDL), PITR
  - HW: multiplexed redo logs, archive logs
  - SW: distribuované systémy
- Redo log
  - každá změna zapsaná **na redolog disk**
  - nezapisují se celá data, jen změny
  - rekonstrukce všech změn provedených v DB
  - před ukončením commit
  - zápis samotných dat asynchronní (vyšší výkon)

Vlastnost: dostupnost 2

- Undo records (undo/rollback segments)
  - odvolání (rollback) transakce
  - vrácení nepotvrzených změn při obnově
  - zajišťuje verzování (původní data)
  - „before image“ u necommitnutých transakcí
    - jiný uživatel čte původní data

Vlastnost: výkon

- Velké objemy dat, velká režie
- Výkon jedné operace x propustnost celku
- Omezení klasické role OS
  - nepoužívat ani cache systému
- Přístup na disk, velké množství dat
  - nejdražší operace
  - vlastní systém cache dat = **sdílená paměť**
    - snížení počtu přístupů
  - přímý přístup k disku (obejít systém souborů)
  - asynchronní (když je čas)
  - disk jen pro DB, žádné soupeření

Vlastnost: výkon 2

- Provádění příkazů
  - prepare, execute, { fetch }, …
    - syntaktická správnost, existence objektů, práva, optimalizace; provádí se jen jednou
  - optimalizace přístupu k datům na úrovni DB serveru
    - efektivní dotazy
  - uložení dat – rychlý zápis x rychlý přístup
    - Indexy – pomocná datová struktura
- In-memory DB
  - využití zejména operační paměti
  - jednodušší algoritmy, takže menší zátěž procesoru
  - výrazně méně drahých I/O operací
  - využití: telekomunikace

Distribuované DBs

- Motivace:
  - vysoká dostupnost, transparentní vůči aplikacím
  - navýšení propustnosti, horizontální zvýšení výkonu
- Shared nothing clustery
  - nezávislé a soběstačné uzly (standby databáze)
    - autonomní ukládání a zpracování dat
  - obrovská škálovatelnost, rozdělení zátěže
  - single point of failure
  - no single point of contention
    - nesdílejí paměť ani disky
  - Google: pure SN, spousta levných počítačů

Distribuované DBs 2

- In-memory databáze
- problémy s distribucí db, katalogu a provádění globálních transakcí
- netriviální množství komunikace navíc

- Shared everything clustery
  - on-line sdílení dat více instancemi (čtení i zápis)
  - global cache: sdílení na úrovni paměti
  - Oracle RAC

Závěr

- Podrobněji v předmětu PV136
- Architektura Oracle Database (obrázek)
- Dotazy
- Příště data warehouse

Architektura Oracle DB

- server: instance a databáze
- instance: SGA a procesy na pozadí
  - SGA: sdílená paměť všemi procesy na pozadí, data a řídící info pro instanci, alokuje se při spuštění instance
  - Procesy: PMON – monitoruje procesy, uvolňuje zdroje; SMON – obnova instance; DBW – zápis změněných dat na disk; LGWR – zapisuje data z redo log bufferu
- databáze: data na disku
- redo log buffer: cachuje info o změnách
- shared pool: library, dictionary a result cache pro paralelní provádění operací

Architektura Oracle DB 2

- PGA: data a řídící info pro konkrétní server
- Database buffer
  - ukládá naposledy použité bloky dat
  - udržuje používané bloky dat v paměti