

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Transakcje w bazach relacyjnych

Sebastian Ernst

Katedra Informatyki Stosowanej AGH

5 grudnia 2013

Outline

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

- 1 Transakcje w bazach relacyjnych
- 2 Poziomy izolacji transakcji
- 3 Wysoka dostępność (HA) w systemach transakcyjnych

Transakcje

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Transakcją nazywamy zbiór operacji, które wykonywane są zgodnie z zasadami ACID:

- *atomicity* – atomiczność,
- *consistency* – spójność,
- *isolation* – izolacja,
- *durability* – trwałość.

Transakcje mogą być izolowane w różnym stopniu. Standard SQL definiuje cztery poziomy izolacji transakcji.

Nie spełnienie zasad izolacji transakcji może prowadzić do *anomalii*. Im niższy poziom izolacji, tym więcej anomalii jest możliwe.

Transakcja rozpoczyna się od polecenia BEGIN i kończy się zatwierdzeniem (COMMIT) lub anulowaniem (ROLLBACK).

Atomiczność

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Transakcja wykonywana jest w całości, bądź wcale – zgodnie z zasadą „wszystko albo nic”. Atomiczność musi być zapewniona także w przypadku awarii zasilania bądź „wysypania się” któregoś z komponentów oprogramowania.

Spójność

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Transakcja musi przeprowadzać bazę z jednego poprawnego stanu do innego poprawnego stanu. Dane w bazie muszą spełniać ograniczenia, poprawnie muszą być wykonane również operacje takie jak kaskady, wyzwacze, itd.

Izolacja

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Ułożenie transakcji w czasie nie powinno mieć wpływu na ich wynik: musi on być taki, jaki by był gdyby transakcje były wykonywane jedna po drugiej.

Trwałość

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Po zatwierdzeniu transakcji, jej efekty muszą być trwałe i odporne na przerwę zasilania, błąd oprogramowania lub systemu.

Implementacja transakcji

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Transakcje mogą być implementowane przez bazy na dwa sposoby:

- **Blokowanie (*locking*):** dane, do których miała dostęp transakcja są oznaczane i SZBD nie pozwala innym transakcjom na ich modyfikację. Pozyskiwanie blokad przy złożonych transakcjach wprowadza narzut i podnosi ryzyko powstawania zakleszczeń.
- **Multiversioning:** SZBD pamięta poprzednie (aktualne przed rozpoczęciem niezatwierdzonej transakcji) wersje danych. Dzięki temu transakcje zmieniające pewne dane nie blokują transakcji, które jednocześnie chcą je odczytywać.

Link: Implementacja MVCC (Multi Version Concurrency Control) w PostgreSQL.

Anomalie

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Słaba izolacja transakcji może prowadzić do następujących anomalii:

- **dirty read** – transakcja może odczytać dane zapisane przez inną, niezatwierdzoną transakcję,
- **nonrepeatable read** – transakcja ponownie odczytuje dane które czytała wcześniej i otrzymuje inne wartości,
- **phantom read** – transakcja ponownie wykonuje zapytanie zwracające zbiór rekordów (np. `WHERE suma > 3000`) i otrzymuje inne rekordy.

Poziomy izolacji transakcji

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

- **Read uncommitted** – dopuszcza wszystkie 3 anomalie,
- **Read committed** (domyślny w PostgreSQL) – dopuszcza //nonrepeatable read// oraz //phantom read//
- **Repeatable read** – dopuszcza //phantom read//,
- **Serializable** – pełna izolacja transakcji.

Poziom transakcji ustawia się przy pomocy polecenia:

```
SET TRANSACTION ISOLATION LEVEL TO serializable;
```

Read Committed

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Zapytanie SELECT widzi tylko dane zatwierdzone przed rozpoczęciem jego wykonania – nie widzi zmian niezatwierdzonych lub zatwierdzonych podczas jego wykonywania.

Zapytania UPDATE, DELETE, SELECT FOR UPDATE, oraz SELECT FOR SHARE w identyczny sposób wyszukują docelowe rekordy. Jeżeli jednak te rekordy są już aktualizowane w innej transakcji, transakcja „kandydująca” poczeka na jej zakończenie.

Repeatable Read

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Na tym poziomie, operacje widzą tylko dane zatwierdzone przed rozpoczęciem transakcji. Polecenia SELECT pracują na obrazie (*snapshot*) bazy utworzonym w momencie jej otwierania.

Aplikacje muszą spodziewać się, że polecenia modyfikujące dane zakończą się z błędem:

```
ERROR: could not serialize access due to concurrent u
```

Trzeba pamiętać, że wyniki transakcji nie muszą tu być identyczne z efektami sekwencyjnego wykonania identycznych transakcji – przykład na kolejnym slajdzie.

Serializable

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

Poziom symuluje sekwencyjne wykonanie transakcji. Wiąże się to z największym ryzykiem powstania zakleszczeń. W PostgreSQL, poziom ten działa identycznie jak Repeatable Read, ale dokonuje dodatkowych sprawdzeń.

Serializable – przykład

Transakcje w bazach relacyjnych

Sebastian Ernst

Transakcje w bazach relacyjnych

Poziomy izolacji transakcji

Wysoka dostępność (HA) w systemach transakcyjnych

Weźmy bazę:

class		value
1		10
1		20
2		100
2		200

i transakcję, która wykonuje

```
SELECT SUM(value) FROM mytab WHERE class = 1;
```

i wstawią wynik z class = 2, oraz analogiczne wykonanie dla przeciwnych klas.

W przypadku Repeatable Read, operacje zostaną wykonane; na poziomie serializable ich wykonanie zakończy się błędem:

```
ERROR: could not serialize access due to read/write d
```

Serializable – dobre praktyki

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

- Oznaczanie transakcji jako `READ ONLY` gdy do możliwe.
- Kontrolowanie liczby aktywnych połączeń z SZBD.
- Definiowanie transakcji tak małych jak to możliwe przy zachowaniu logicznej integralności.
- Nie pozostawianie połączeń oczekujących (idle) w środku transakcji.
- Ograniczenie jawnych blokad – nie są potrzebne, bo integralność zabezpieczają mechanizmy `SERIALIZABLE`.
- Skanowanie sekwencyjne wymaga blokady predykatu na poziomie tabeli – użycie indeksów może ograniczyć liczbę błędów serializacji.

Jawne blokowanie (explicit locking)

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

PostgreSQL pozwala na ręczne nakładanie blokad na wybrane obiekty BD.

Blokady tabel są nakładane automatycznie przez niektóre polecenia, można też je wymusić ręcznie przy pomocy polecenia LOCK. Istnieje kilka trybów blokowania, dla których określono relację *konfliktu*. Dwie transakcje nie mogą jednocześnie posiadać konfliktujących blokad na tej samej tabeli.

Blokady rekordów mogą być wyłączne (exclusive) lub dzielone (shared). Blokada wyłączna jest automatycznie zakładana przy aktualizacji lub usuwaniu rekordu. Do pobrania blokady wykluczającej służy polecenie SELECT FOR UPDATE.

Drugim typem blokady rekordu jest blokada dzielona. Kilka transakcji może posiadać blokadę dzieloną na danym rekordzie jednocześnie, jednak żadna nie może wtedy wykonywać na nim operacji UPDATE lub DELETE, lub uzyskać blokady wyłącznej.

Zestawienie rozwiązań HA

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

- **Shared Disk Failover (Przełączanie ze współdzielonym dyskiem).** Jedna macierz dyskowa jest współdzielona przez kilka serwerów. W przypadku awarii serwera, serwer zapasowy (standby) montuje system plików i przejmuje świadczenie usług.
- **File System Replication (Replikacja systemu plików).** Wszystkie zmiany w urządzeniu blokowym (systemie plików) są odzwierciedlane na urządzeniach zapasowych. Przykład rozwiązania: DRBD.
- **Warm and Hot Standby using Point-in-Time Recovery (PITR).** Serwery zapasowe są utrzymywane w aktualnym stanie poprzez strumień rekordów WAL (Write-Ahead Log). W przypadku awarii serwera głównego, serwer zapasowy ma prawie wszystkie jego dane.

Zestawienie rozwiązań HA (c.d.)

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

- **Trigger-Based Master-Standby Replication (Replikacja master-slave).** Serwer główny (master) asynchronicznie wysyła zmiany danych do serwera rezerwowego (standby). Serwer rezerwowy może odpowiadać na zapytania typu read-only, nadaje się więc idealnie do zastosowań typu hurtownia danych. Przykład: Slony-I.
- **Statement-Based Replication Middleware (Replikacja oparta o zapytania).** Dodatkowe oprogramowanie przechwytuje zapytania i rozsyła do wybranych serwerów. Zapytania modyfikujące dane wysyłane są do wszystkich serwerów; zapytania read-only – tylko do jednego. Należy dopilnować poprawności wykonania transakcji na wszystkich serwerach np. poprzez zatwierdzanie dwufazowe (PREPARE TRANSACTION/COMMIT PREPARED) Przykład: Pgpool-II.

Zestawienie rozwiązań HA (c.d.)

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

- **Asynchronous Multimaster Replication (Asynchroniczna replikacja z równorzędnymi serwerami)**. Stosowana, gdy serwery komunikują się nieregularnie/co jakiś czas. Konflikty synchronizacji rozwiązywane przy pomocy odpowiednich reguł. Przykład: Bucardo.
- **Synchronous Multimaster Replication (Synchroniczna replikacja z równorzędnymi serwerami)**. Każdy serwer może przyjąć zapytanie modyfikujące dane, a zmiany są propagowane na pozostałe serwery przed zakończeniem transakcji. Bardzo słaba wydajność operacji zapisu.

Link: Porównanie metod

Inne rozwiązania

Transakcje w
bazach
relacyjnych

Sebastian
Ernst

Transakcje w
bazach
relacyjnych

Poziomy
izolacji
transakcji

Wysoka
dostępność
(HA) w
systemach
transakcyjnych

- **Partycjonowanie danych.** Podział atrybutów między kilka tabel, bądź tabel między kilka baz.
- **Sharding.** Podział danych ze względu na wartości atrybutów (np. oddział firmy, pierwsza litera nazwiska, itd.). Każdy zbiór jest modyfikowany przez jeden określony serwer.
- **Równoległe wykonywanie zapytań na kilku serwerach.** Skomplikowane zapytania wykonywane są na kilku serwerach równoległe w celu zwiększenia wydajności. Dane są dzielone między serwery; te przetwarzają podzbiory danych i przekazują do centralnego serwera w celu scalenia. Funkcjonalność dostępna w Pgpool-II.

Link: Dlaczego Instagram nie przeszedł na NoSQL.

Link Shard Lessons