



Bazy Danych

Andrzej M. Borzyszkowski
Instytut Informatyki

Uniwersytetu Gdańskiego

materiały dostępne elektronicznie

<http://inf.ug.edu.pl/~amb>

© Andrzej M. Borzyszkowski

Bazy Danych

3/17

Cele i narzędzia zarządzania współbieżnością

- Szeregowalność
 - dopuszczenie współbieżności jest pożądane/konieczne z powodu wydajności
 - sprawdzanie szeregowalności nie jest słuszne
 - przebieg nie jest znany z góry, wynika ze stanu obliczeń
 - raczej należy gwarantować szeregowalność przebiegu transakcji
- Cel: ten sam wynik zmian w bazie danych
- Narzędzia
 - blokady
 - znaczniki czasu
 - wielowersyjność
 - protokoły optymistyczne

© Andrzej M. Borzyszkowski

Bazy Danych

Blokady i inne narzędzia zarządzania współbieżnością

© Andrzej M. Borzyszkowski

Bazy Danych

2/17

Blokady binarne

- Obiekt X może być zablokowany
 - transakcja ma prawo blokować tylko obiekt nieblokowany
 - może być utworzona kolejka transakcji do blokowania X
 - transakcja może odblokować obiekt, który zablokowała
 - protokół wzajemnego wykluczania – co najwyżej jedna transakcja blokuje obiekt
- Blokowanie jeszcze nie gwarantuje szeregowalności przebiegu
 - $l1(Y); r1(Y); u1(Y); l2(X); r2(X); u2(X); l2(Y); r2(Y); w2(Y); u2(Y); l1(X); r1(X); w1(X); u1(X);$
- te blokady niczego nie dają, są za wcześnie zwalniane
 - $r1(Y); r2(X); r2(Y); w2(Y); r1(X); w1(X);$ wymaga $1 < 2 < 1$
 - blokowanie binarne być może blokuje za dużo

© Andrzej M. Borzyszkowski

Bazy Danych

4/17

Blokady czytania i zapisu

- blokada współdzielona (shared) - blokada do odczytu
- blokada wyłączna (exclusive) - blokada do zapisu
- Transakcja czytająca dane musi założyć blokadę współdzieloną, wiele transakcji może założyć taką blokadę, nie można jej założyć, jeśli jest już blokada wyłączna
- Transakcja zapisująca dane musi założyć blokadę wyłączną, nie można jej założyć, jeśli jest już założona jakakolwiek blokada

nowa/dotychczasowa	X	S	brak
X	nie	nie	tak
S	nie	tak	tak

- Różne wersje: jedno zwolnienie blokady, zwolnienie blokady zapisu pozostawiając blokadę odczytu, brak możliwości podnoszenia stopnia blokady

5/17

© Andrzej M. Borzyszkowski

Bazy Danych

Współbieżność vs. blokady

- Niespójna analiza:

czas	użytkownik 1	użytkownik 2
0 min	blokada czytania konta A wynik 100	
1 min		próba blokady zapisu A nieudana, trzeba czekać
2 min	blokada czytania konta B wynik 100	
3 min	zwolnienie blokad	
4 min		blokada zapisu konta A i dalszy ciąg transakcji

6/17

© Andrzej M. Borzyszkowski

Bazy Danych

Współbieżność vs. blokady

- Niespójna analiza w innej kolejności:

czas	użytkownik 1	użytkownik 2
0 min	blokada czytania konta A wynik 100	
1 min		blokada zapisu konta B dodaje 50 do konta B
2 min		blokada zapisu konta A nieudana, trzeba czekać
3 min	blokada czytania konta B nieudana, trzeba czekać	

- te transakcje czekają na siebie nawzajem - zakleszczenie

7/17

© Andrzej M. Borzyszkowski

Bazy Danych

Współbieżność vs. blokady, c.d.

- Utracona modyfikacja:

czas	użytkownik 1	użytkownik 2
0 min	blokada czytania konta A	
1 min		blokada czytania konta A
2 min	blokada zapisu konta A nieudana, trzeba czekać	
3 min		blokada zapisu konta A nieudana, trzeba czekać

- znowu zakleszczenie (deadlock)

8/17

© Andrzej M. Borzyszkowski

Bazy Danych

Zakleszczenie, rozwiązanie 1

- Rozwiązanie 1: timeout
 - system zarządzania bazą danych wycofuje transakcję, która zbyt długo oczekiwała na zwolnienie blokady
 - czy transakcja będzie powtórzona?
 - w PostgreSQL nieautomatycznie, może to zrobić aplikacja działająca w pętli aż do pozytywnego zakończenia transakcji
- Wersje:
 - zero czekania – transakcja żądająca niemożliwej blokady jest natychmiast wycofywana
- Problem zagłodzenia
 - nie ma gwarancji, że wycofana transakcja doczeka się wykonania
 - konieczne są metody promocji transakcji wycofywanych

© Andrzej M. Borzyszkowski

Bazy Danych

9/17

Zakleszczenie, rozwiązanie 2,3

- Rozwiązanie 2: analiza grafu oczekiwań
 - SZBD analizuje graf wzajemnych oczekiwań na zwolnienie blokady i wycofuje jedną z transakcji
 - np. najnowszą, lub najstarszą, lub najmniejszą, lub najmniej ważną, lub
 - problem zagłodzenia jest obecny, ta sama transakcja nie powinna być ciągle wycofywana
- Rozwiązanie 3: wszystkie blokady powinny być zakładane w tej samej kolejności – wówczas nie będzie zakleszczeń

© Andrzej M. Borzyszkowski

Bazy Danych

10/17

Zakleszczenie, rozwiązanie 4

- Rozwiązanie 4: znaczniki czasu dla transakcji
 - wersja „czekaj albo zgiń”: transakcja starsza może czekać, młodsza jest wycofywana i powtórnie wykonana z tym samym znacznikiem
 - wersja „zabij albo czekaj”: transakcja starsza powoduje wycofanie młodszej (i wykonanie z tym samym znacznikiem), transakcja młodsza czeka
 - na pewno nie będzie zakleszczenia
 - wycofywane są transakcje, które być może nie powodują w ogóle zakleszczenia

© Andrzej M. Borzyszkowski

Bazy Danych

11/17

Blokowanie dwufazowe (2PL)

- Protokół blokowanie dwufazowego:
 - faza 1: transakcja zakłada potrzebne blokady (rozszerzanie)
 - faza 2: transakcja zwalnia blokady (kurczenie)
- Twierdzenie: jeśli wszystkie transakcje przestrzegają protokołu blokowania dwufazowego, to dowolny przebieg jest szeregowalny
 - ale zwiększa to niebezpieczeństwo zakleszczenia

© Andrzej M. Borzyszkowski

Bazy Danych

12/17

Blokowanie dwufazowe – wersje

- Blokowanie statyczne – transakcja z góry określa swoje blokady
 - jeśli nie może założyć wszystkich, to czeka
 - na pewno nie wystąpi zakleszczenie
 - protokół mało praktyczny, nie zawsze znane są potrzeby
 - transakcja może nigdy nie doczekać się wykonania
- Blokowanie ściśle – transakcja zwalnia blokady zapisu dopiero na końcu
 - blokowanie rygorystyczne – transakcja zwalnia wszystkie blokady dopiero na końcu
 - gwarantowany jest przebieg ścisły (szeregowalność, łatwe odtwarzanie)
 - możliwość zakleszczeń

© Andrzej M. Borzyszkowski

Bazy Danych

13/17

Inne narzędzia rozwiązania problemów współbieżności

- Znaczniki czasu
 - każda transakcja ma swój znacznik czasu
 - każdy obiekt ma zapisany czas ostatniego odczytu i zapisu przez transakcje jeszcze nie zatwierdzone
 - różne algorytmy wycofujące transakcje, które mogłyby zagrozić pojęciu szeregowalności
- Wielowersyjność
 - SZBD utrzymuje wiele wersji bazy danych dla niezatwierdzonych transakcji, odczyty i zapisy dotyczą odpowiednich wersji
- Techniki optymistyczne
 - transakcje są wykonywane bez przeszkód
 - przy zatwierdzaniu transakcji zaczyna się sprawdzanie, czy mogło dojść do naruszenia spójności bazy danych

© Andrzej M. Borzyszkowski

Bazy Danych

15/17

Blokady a SQL

- SQL nie przewiduje w ogóle jawnego zakładania i zwalniania blokad
 - blokada czytania wierszy będzie założona jeśli użyje się instrukcji **SELECT 1 FROM ____ WHERE ____** zmuszając system do czytania wierszy
 - blokada zapisu wierszy będzie założona jeśli użyje się instrukcji **SELECT 1 FROM ____ WHERE ____ FOR UPDATE** anonsując chęć zapisu w wierszach
- PostgreSQL dopuszcza jeszcze instrukcję **LOCK TABLE ____** nie należy tej możliwości nadużywać, bo ma poważne konsekwencje dla wydajności
- SQL nie daje możliwości blokowania poszczególnych atrybutów

© Andrzej M. Borzyszkowski

Bazy Danych

14/17

Znaczniki czasu

- Obiekty mają czasy ostatniego jeszcze niezatwierdzonego odczytu timeR i zapisu timeW
- Przykład: transakcja T czyta X
 - jeśli $time(T) < timeW(X)$ to znaczy, że transakcja chce czytać obiekt już zmieniony – wycofanie T
 - jeśli $time(T) > timeW(X)$ to jest OK, T czyta X i $timeR(X)$ jest być może powiększony do $time(T)$
- Przykład: transakcja T zapisuje X
 - jeśli $time(T) < timeR(X)$ to znaczy, że obiekt był już odczytany później jak gdyby T nie istniało – wycofanie T
 - jeśli $time(T) < timeW(X)$ to znaczy, że obiekt był już zapisany później jak gdyby T nie istniało – wycofanie T
 - Wpp jest OK, T zapisuje X i $timeW(X) = time(T)$

© Andrzej M. Borzyszkowski

Bazy Danych

16/17

Znaczniki czasu, c.d.

- Przebiegi są na pewno szeregowalne, mamy ustaloną kolejność transakcji
- Nie ma zakleszczeń, bo żadne transakcje nie czekają na siebie, najwyżej są wycofywane
- Przebiegi mogą powodować kaskady
- Przebiegi mogą być nieodtworzalne
- Rozwiązania:
 - nie czyta się danych niezatwierdzonych
 - albo analizuje kolejność zatwierdzeń

© Andrzej M. Borzyszkowski

Bazy Danych