Введение в реляционные базы данных

Лекция 9: Транзакции

Артем Толканев

December 06, 2024

Транзакции

Есть последовательности инструкций запроса и/или изменения данных.

Набор этих операций, которые можно считать в конкретном случае логически одной, то есть полностью произойдут (**COMMIT**) или полностью отменятся (**ROLLBACK**), называют **транзакцией**.





Atomicity (Атомарность)

Либо фиксирует действия транзакции после завершения всех ее этапов, либо откатывает все ее действия в случае, если транзакция не смогла успешно выполнить все свои действия.

Consistency (Согласованность)

После выполнения транзакции база данных остается "адекватной".

Isolation (Изоляционность)

Во время работы транзакции другие транзакции не должны оказывать влияние на результаты её работы.

Durability (Устойчивость)

Внезависимости от того, что где-либо что-то происходит - если мы получили Ок после выполнения транзакции, то это действительно Ок - все изменения произведены даже если будет сбой.





Соответственно возможны аномалии:

Read-Write (R-W) (Unrepeatable Read)

Write-Read (W-R)

Write-Write (W-W)

T1 BEGIN R(A) = 2000	T2 BEGIN
	R(A) = 2000 W(A) = 3000 COMMIT
R(A) = 3000 COMMIT	

Соответственно возможны аномалии:

Read-Write (R-W)

Write-Read (W-R) Dirty Read

Write-Write (W-W)

T1 BEGIN R(A) = 2000	T2 BEGIN
W(A) = 3000	R(A) = 3000 W(A) = 3000 - 500 COMMIT
ROLLBACK	

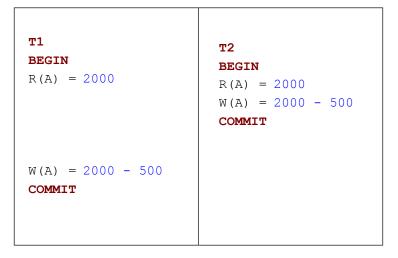


Соответственно возможны аномалии:

Read-Write (R-W)

Write-Read (W-R)

Write-Write (W-W) Lost Update



SERIAL

CREATE TABLE test(id serial, value int);

identity(1,1)





Synopsis

CREATE [[GLOBAL | LOCAL] { TEMPORARY | TEMP } | UNLOGGED] TABLE [IF NOT EXISTS] table_name ([column_name data_type [STORAGE { PLAIN | EXTERNAL | LATENDED | MAIN | DEFAULT }] [COMPRESSION compre

Is a Postgresql UNLOGGED table completely lost on process crash?

Asked 1 year, 8 months ago Modified 1 year, 8 months ago Viewed 629 times



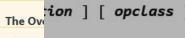
I use UNLOGGED tables for a few very large tables in a data warehouse style application.



Until recently, I understood UNLOGGED to mean "won't write to the WAL" - which in turn means that *recent* changes maybe lost on a process crash / unclean termination, and that there will be no replication



Maybe I'm misunderstanding the language in the documentation (or maybe I'm not) but when I















IMMEDIATE

DEFERRABLE

CONSTRAINTS:



UNIQUE,

PRIMARY KEY,

REFERENCES (foreign key), EXCLUDE

отсроченные





мгновенные





IMMEDIATE

DEFERRABLE

CONSTRAINTS:



UNIQUE,

PRIMARY KEY,

REFERENCES (foreign key), EXCLUDE

отсроченные

ALTER TABLE ... ADD CONSTRAINT ... UNIQUE (...) DEFERRABLE





LOCK

T1 BEGIN W(A) = 2000 - 100 W(B) = 2000 + 100 COMMIT T2 BEGIN W(B) = 2000 + 50 W(A) = 2000 - 50 COMMIT





LOCK

Транзакции запрашивают блокировки на данные.

Диспетчер блокировок удовлетворяет или блокирует запросы.

Транзакции снимают блокировки.

Диспетчер блокировок обновляет свою внутреннюю таблицу блокировок.

```
T1
BEGIN
W(A) = 2000 - 100
W(B) = 2000 + 100
COMMIT

T2
BEGIN
W(B) = 2000 + 50
W(A) = 2000 - 50
COMMIT
```



