

**Транзакции. Механизм блокировок.  
Виды блокировок. Проблемы  
применения**

Выполнил:

# База данных

- ▶ База данных — представленная в объективной форме совокупность самостоятельных материалов (статей, расчётов, нормативных актов, судебных решений и иных подобных материалов), систематизированных таким образом, чтобы эти материалы могли быть найдены и обработаны с помощью электронной вычислительной машины (ЭВМ).



# Транзакции

Любая база данных годна к использованию только тогда, когда ее состояние соответствует состоянию предметной области. Такие состояния называют целостными. Очевидно, что при изменении данных БД должна переходить от одного целостного состояния к другому. Однако, в процессе обновления данных возможны ситуации, когда состояние целостности нарушается.

Во избежание таких ситуаций в СУБД вводится понятие транзакции - атомарного действия над БД, переводящего ее из одного целостного состояния в другое целостное состояние. Другими словами, транзакция - это последовательность операций, которые должны быть или все выполнены или все не выполнены (все или ничего).



# Транзакции

Описанный механизм транзакций гарантирует обеспечение целостного состояния базы данных только в том случае, когда все транзакции выполняются последовательно, т.е. в каждую единицу времени активна только одна транзакция. В то же время, если одновременно выполняются две транзакции, могут возникнуть следующие ошибочные ситуации:

- ▶ Грязное чтение (Dirty Read) - транзакция T1 модифицировала некий элемент данных. После этого другая транзакция T2 прочитала содержимое этого элемента данных до завершения транзакции T1
- ▶ Неповторяемое (размытое) чтение (Non-repeatable or Fuzzy Read) - транзакция T1 прочитала содержимое элемента данных. После этого другая транзакция T2 модифицировала или удалила этот элемент.
- ▶ Фантом (фиктивные элементы) (Phantom) - транзакция T1 прочитала содержимое нескольких элементов данных, удовлетворяющих некому условию. После этого T2 создала элемент данных, удовлетворяющий этому условию и зафиксировалась.

# Механизм блокировок

- ▶ Принудительное упорядочение транзакций обеспечивается с помощью механизма **блокировок**. Суть этого механизма в следующем: если для выполнения некоторой транзакции необходимо, чтобы некоторый объект базы данных (кортеж, набор кортежей, отношение, набор отношений,..) не изменялся непредсказуемо и без ведома этой транзакции, такой объект блокируется. Основными видами блокировок являются:
- ▶ **блокировка со взаимным доступом**, называемая также *S-блокировкой* (от Shared locks) и *блокировкой по чтению*.
- ▶ **монополярная блокировка** (без взаимного доступа), называемая также *X-блокировкой* от (eXclusive locks) или *блокировкой по записи*. Этот режим используется при операциях изменения, добавления и удаления объектов.



# Механизм блокировок

- ▶ Транзакция, запросившая доступ к объекту, уже захваченному другой транзакцией в несовместимом режиме, останавливается до тех пор, пока захват этого объекта не будет снят.
- ▶ Доказано, что сериализуемость транзакций (или, иначе, их изоляция) обеспечивается при использовании двухфазного протокола блокировок (2LP - Two-Phase Locks), согласно которому все блокировки, произведенные транзакцией, снимаются только при ее завершении. Т.е выполнение транзакции разбивается на две фазы: (1) - накопление блокировок, (2) - освобождение блокировок в результате фиксации или отката.

# Проблемы применения

К сожалению, применение механизма блокировки приводит к замедлению обработки транзакций, поскольку система вынуждена ожидать пока освободятся данные, захваченные конкурирующей транзакцией. Решить эту проблему можно за счет уменьшения фрагментов данных, захватываемых транзакцией. В зависимости от захватываемых объектов различают несколько уровней блокировки:

- ▶ блокируется вся база данных - очевидно, этот вариант неприемлим, поскольку сводит многопользовательский режим работы к однопользовательскому
- ▶ блокируются отдельные таблицы
- ▶ блокируются страницы (страница - фрагмент таблицы размером обычно 2-4 Кб, единица выделения памяти для обработки данных системой)
- ▶ блокируются записи
- ▶ блокируются отдельные поля

# Проблемы применения

- ▶ Одним из наиболее серьезных недостатков метода сериализации транзакций на основе механизма блокировок является возможность возникновения тупиков (dead locks) между транзакциями. Пусть, например, транзакция T1 наложила монопольную блокировку на объект O1 и претендует на доступ к объекту O2, который уже монопольно заблокирован транзакцией T2, ожидающей доступа к объекту O1. В этом случае ни одна из транзакций продолжаться не может, следовательно, блокировки объектов O1 и O2 никогда не будут сняты. Естественного выхода из такой ситуации не существует, поэтому тупиковые ситуации обнаруживаются и устраняются искусственно. При этом СУБД откатывает одну из транзакций, попавших в тупик ("жертвует" ею), что дает возможность продолжить выполнение другой транзакции.





# Список использованной литературы

- ▶ К.Дж.Дейт Введение в системы баз данных (6 изд.). Киев, Диалектика, 1998 (главы 13 и 14)
- ▶ С.Д.Кузнецов Введение в СУБД (главы 9 и 10), СУБД N 3, 1996, с.136-144.

**Презентация окончена**

**Спасибо за внимание**