

Последний слайд лекции 4

В общем случае целостность БД может быть проверена лишь по исполнению некоторой логически завершённой последовательности операций обновления (**транзакции**).

Пример ТРАНЗАКЦИИ

Перевод денег с одного банковского счёта на другой

- | | |
|---|-------------|
| а) извлечь запись счёта-источника | (RETRIEVE), |
| б) уменьшить его остаток на переводимую сумму S | (UPDATE), |
| в) извлечь запись счёта-приёмника | (RETRIEVE), |
| г) увеличить его остаток на сумму S | (UPDATE). |

Только после операции г) состояние счетов будет согласованным.

4.4. Транзакции в базе данных

4.4.1. Понятие транзакции

Промежуточные состояния БД
могут быть несогласованными

Транзакция

Последовательность действий над базой данных, выполняемая по запросам одного пользователя или приложения и переводящая БД *из согласованного начального* состояния *в согласованное конечное* состояние.

Может быть представлена

отдельной программой

частью программы

последовательностью операторов ЯМД

отдельным оператором

Транзакция есть *неделимая единица работы* в БД.

4.4.2. Способы завершения транзакции

ФИКСАЦИЯ

COMMIT TRANSACTION

Произведённые транзакцией обновления данных становятся частью БД.

ROLLBACK TRANSACTION

Произведённые транзакцией обновления данных отменяются.
БД возвращается в исходное состояние.

ОТКАТ

4.4.3. Свойства транзакции (свойства АСИД)

Атомарность

Гарантирована **неделимость** транзакции.
Либо выполняются ВСЕ её операции,
либо не выполняется НИ ОДНА.

Согласованность

Гарантирована **согласованность** состояния БД
к моменту начала транзакции,
и после её завершения.

Изолированность

Гарантирована **эквивалентность**
результатов **параллельного** исполнения
любого набора транзакций
результатам **последовательного** исполнения
того же набора
в любой последовательности.

Долговечность

Гарантировано **сохранение** результатов работы
транзакции, **завершившейся фиксацией**.

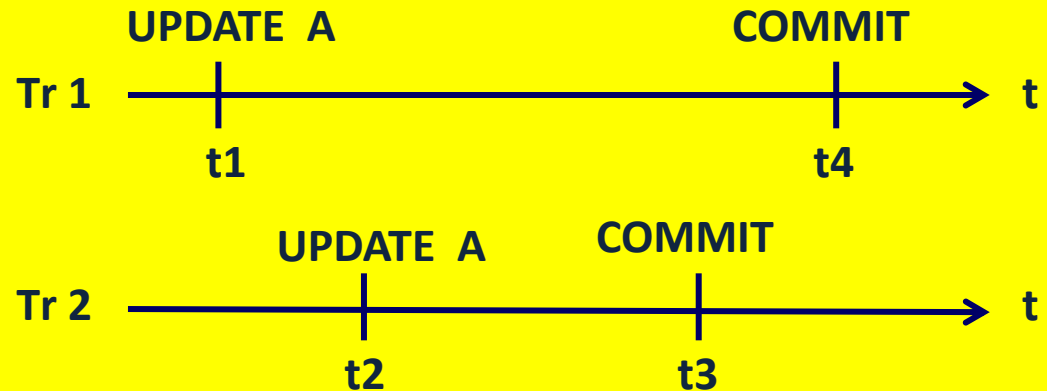
4.5. Управление параллелизмом

Параллелизм

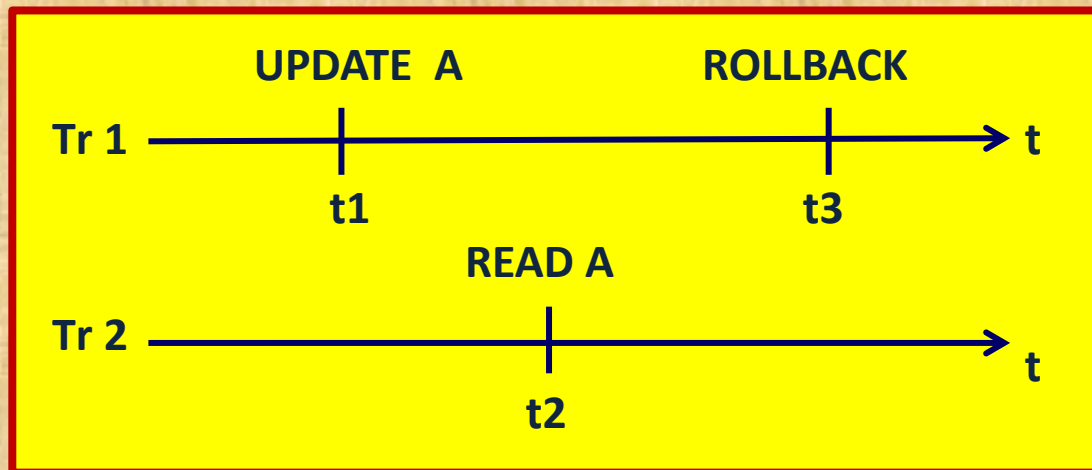
Одновременный доступ к БД нескольких приложений.

4.5.1. Конфликты параллельного доступа (примеры)

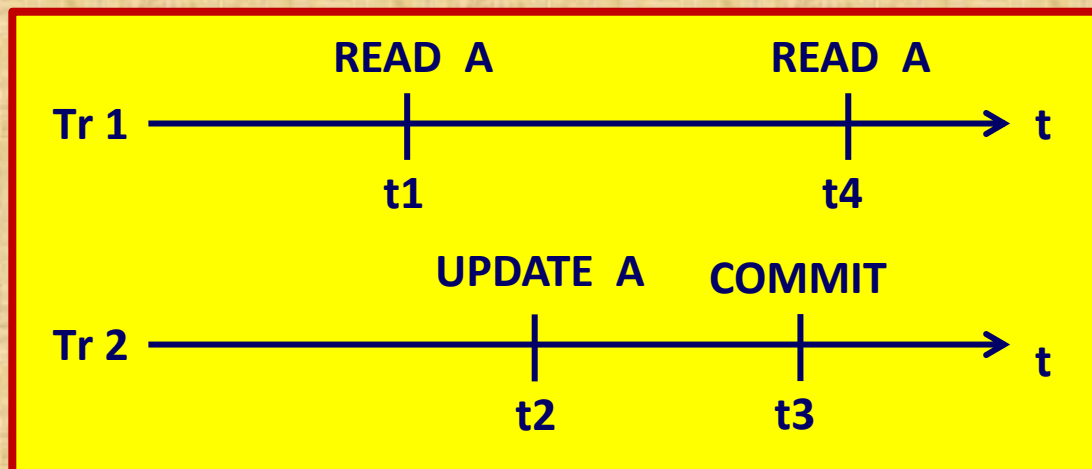
Потеря обновлений



Чтение “грязных” данных



Неповторяющиеся чтения

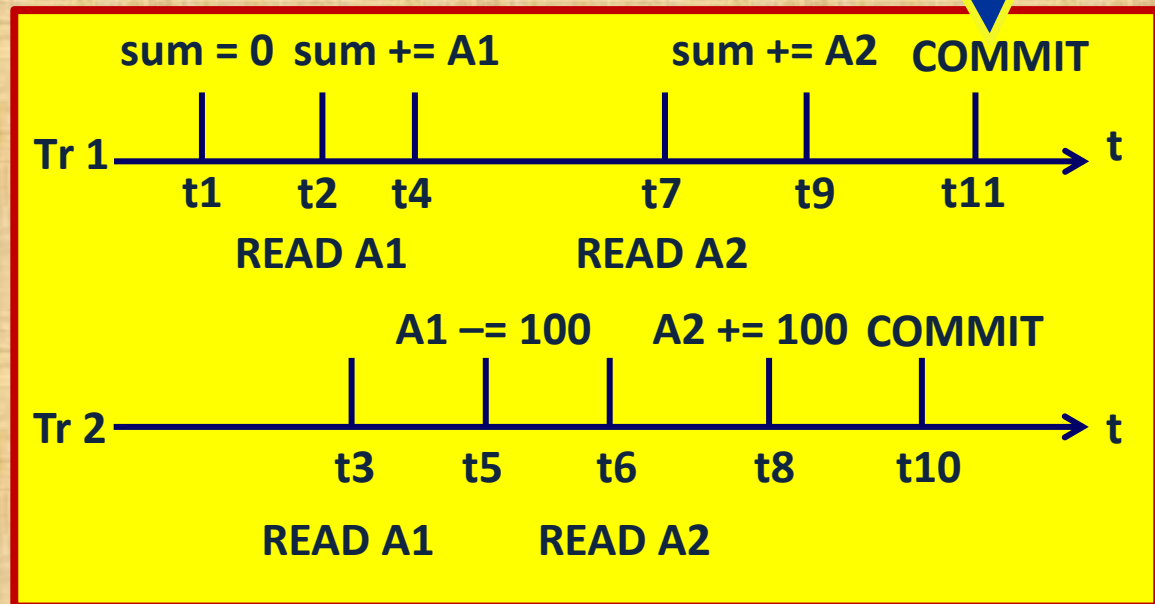


Tr 1 подсчитывает сумму денег на счетах A1, A2, A3,...

Tr 2 переводит деньги со счёта A1 на счёт A2.

Зафиксировано
НЕВЕРНОЕ значение sum

Несогласованная обработка



Подробнее

<http://citforum.ru/database/dblearn/> Гл. 9, 10

4.5.2. Типы конфликтов параллельного доступа

**Запись – Запись
(W-W)**

Транзакция **T1** изменила объект и не закончилась.
Транзакция **T2** изменяет этот объект.

Результат - потеря обновления.

**Чтение–Запись
(R-W)**

Транзакция **T1** прочитала объект и не закончилась.
Транзакция **T2** изменяет этот объект.

Результат - неповторяемое считывание.

**Запись – Чтение
(W-R)**

Транзакция **T1** изменила объект и не закончилась.
Транзакция **T2** читает этот объект.

Результат - чтение "грязных" данных.

**Конфликты типа R-R (Чтение - Чтение) отсутствуют,
т.к. данные при чтении не изменяются.**

4.5.3. Уровни изолированности транзакций

**СУБД должна
гарантировать!**

Первый

Невозможность потери обновлений.

До завершения транзакции, изменяющей объект А никакая другая транзакция не может его изменять.

Второй

Невозможность чтения “грязных данных”.

До завершения транзакции, изменяющей объект А никакая другая транзакция не может его читать.

Третий

Отсутствие неповторяющихся чтений.

До завершения транзакции, читающей объект А никакая другая транзакция не может его изменять.

4.5.4. Сериализация транзакций

Режимы блокировок

X - eXclusive

S - Shared

Совместимость захватов

	X	S
X	Нет	Нет
S	Нет	Да

Фаза наложения блокировок

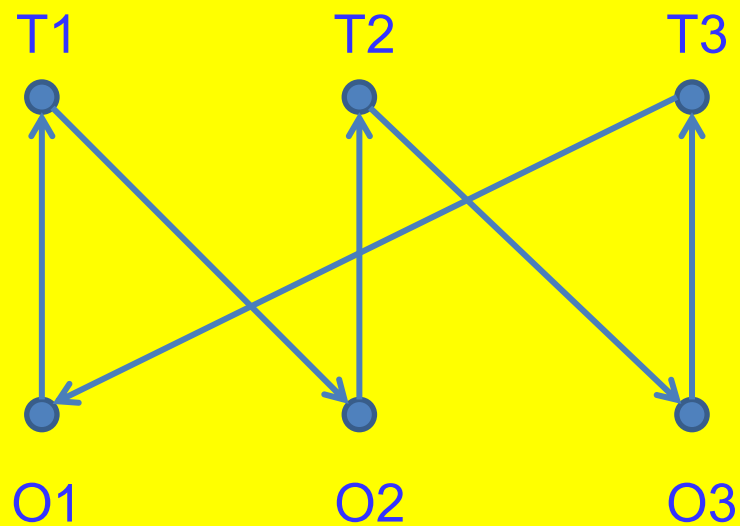
Перед выполнением любой операции над объектом **R** транзакция должна запросить блокировку **R** в подходящем режиме.

Фаза снятия блокировок

Если транзакция сняла хотя бы одну наложенную ранее блокировку, то она не может запрашивать новые.

**Соблюдение двухфазного протокола
гарантирует отсутствие конфликтов типов W-W, R-W и W-R.**

Взаимные блокировки транзакций



4.6. Восстановление базы данных

4.6.1. Типы аварийных ситуаций

Локальный сбой

Аварийное или принудительное прекращение
ОДНОЙ транзакции.

Мягкий сбой

Аварийное прекращение ВСЕХ транзакций

Жёсткий сбой

Физическое разрушение базы данных

**НЕОБХОДИМО
ВОССТАНОВЛЕНИЕ БД**

4.6.2. Системный журнал (журнал транзакций)

Служебный файл. Содержит информацию, необходимую для восстановления БД.

О каждой транзакции

идентификатор транзакции

идентификатор инициатора

время начала

способ завершения

время завершения

О каждой операции

идентификатор транзакции

идентификатор объекта обработки

тип операции

время начала операции

копия объекта до операции

копия объекта после операции

**UPDATE и
INSERT**

**UPDATE и
DELETE**

4.6.3. Правила ведения системного журнала

1. Записи журнала создаются в специальном *буфере журнала*.
2. Каждая запись имеет временную метку.
3. Буфер журнала принудительно выталкивается во внешнюю память при завершении транзакции.

Протокол WAL
(Write Ahead Log)

Результаты зафиксированной транзакции
не могут попасть в ФБД раньше,
чем соответствующие ей записи журнала.

4.6.4. Восстановление после локального сбоя

Сводится к **отмене изменений** в рабочем буфере СУБД, произведённых прерванной транзакцией.

Действия

Выбрать из журнала все записи от прерванной транзакции.

Расположить их в порядке, обратном хронологическому.

Выполнить в рабочем буфере обратные по смыслу операции.

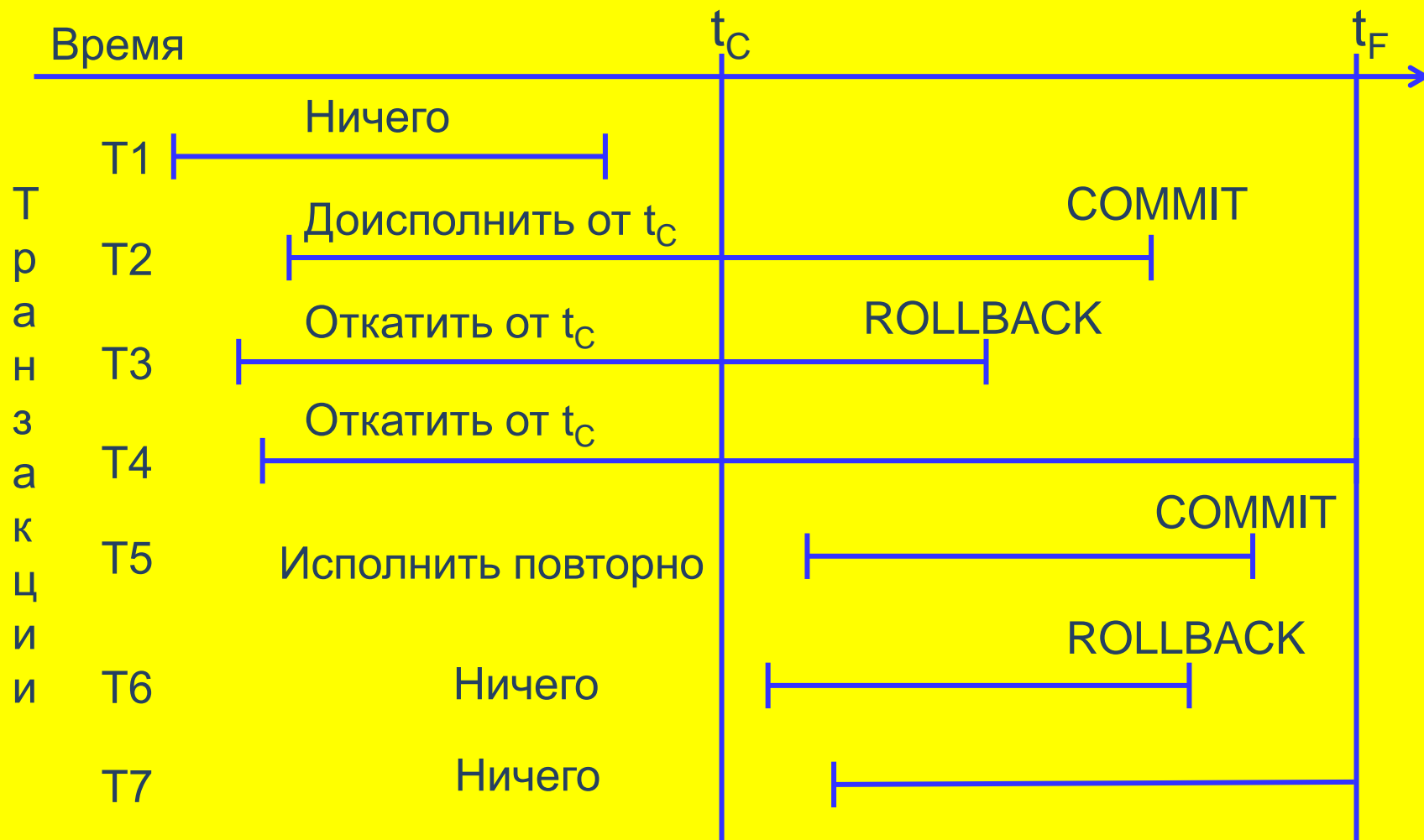
**Процесс восстановления является транзакцией.
Протоколируется в системном журнале.**

4.6.5. Восстановление после мягкого сбоя системы

Потеряно содержимое рабочего буфера СУБД.
Восстановление возможно, если периодически сохраняется состояние буфера «КАК ЕСТЬ» (**Контрольная точка**).

Контрольная точка

Мягкий сбой



При перезагрузке системы СУБД должна

- откатить все прерванные в момент сбоя транзакции и
- выполнить повторно все успешно завершившиеся, но не зафиксированные в ФБД.

Основа для восстановления –

состояние буфера на момент принятия контрольной точки.

4.6.5. Восстановление после жёсткого сбоя

Возможно при наличии **резервной копии ФБД**.

Создание резервной копии

по достижении “жёлтой зоны” в файле журнала.

Действия СУБД

- Запуск новых транзакций не производить.
- Дождаться завершения всех существующих транзакций.
- Вытолкнуть рабочие буферы журнала и БД во внешнюю память.
- Скопировать состояние ФБД на резервный носитель.
- Очистить файл журнала.

Восстановление

- Восстановить состояние ФБД на момент последнего копирования.
- По текущему журналу повторно исполнить все транзакции, успешно завершившиеся до момента сбоя.