

Skillbox

# Управление транзакциями

**Андрей Гордиенков**

Solution Architect

ABAX

# В прошлом и в этом уроке

## В прошлом уроке:

- модели согласованности данных
- производительность и данные

## В этом уроке:

- аномалии данных — ACID и изоляция данных

Skillbox

# Изоляция данных и аномалии

**Андрей Гордиенков**

Solution Architect

ABAX

# Согласованность

BASE-системы не гарантируют строгой согласованности.

ACID — старается, но не всегда получается.

Однако в рамках одной реплики оба подхода полагаются на изоляцию транзакций в процессе внесения изменений.

# Изоляция

Основные уровни изоляции для баз данных:

- Анархия
- Read Uncommitted
- Read Committed
- Repeatable Read / Snapshot Isolation
- Serializable

# Аномалии

Вместе с уровнями изоляции определяются типы аномалий чтения данных.

	Dirty reads	Lost updates	Fuzzy reads	Phantoms	Read skews	Write skews
Read uncommitted						
Read committed						
Repeatable read						
Snapshot						
Serializable						

# Read uncommitted

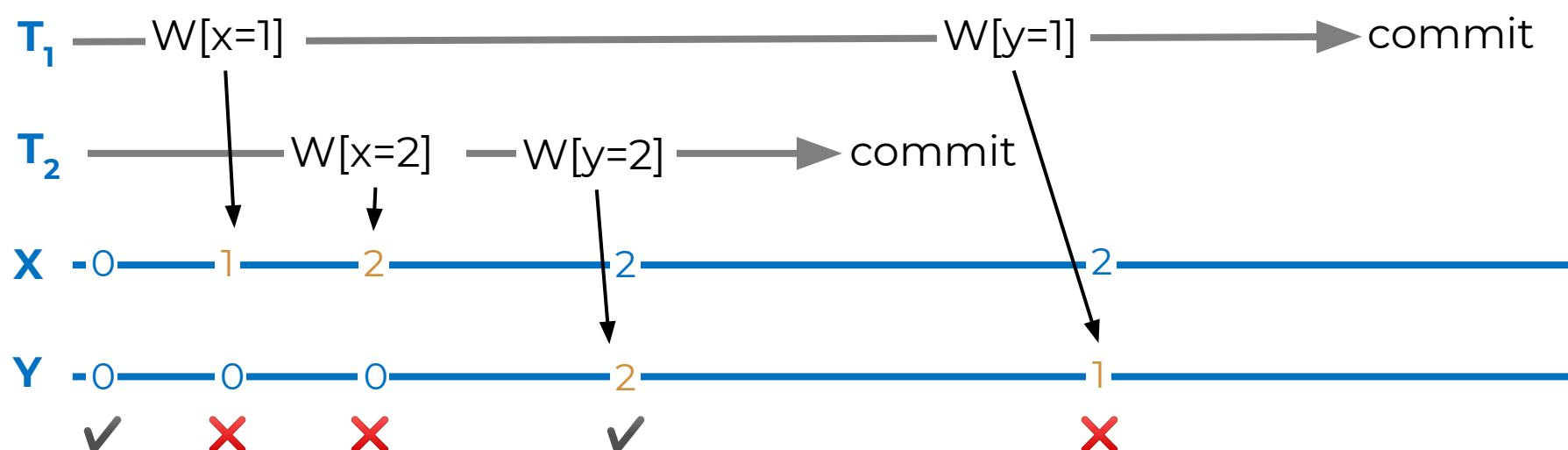
Самый слабый уровень изоляции.

Гарантии: транзакции не перезаписывают данные друг друга в процессе исполнения (no dirty writes).

Иногда используется как «совет» запросу БД не использовать блокировку на чтение для ускорения работы. Но в итоге может быть весь спектр аномалий.

# Dirty Writes

Частичная перезапись данных, сделанных другой транзакцией.



**Ограничение:  $x = y$**



# Read committed

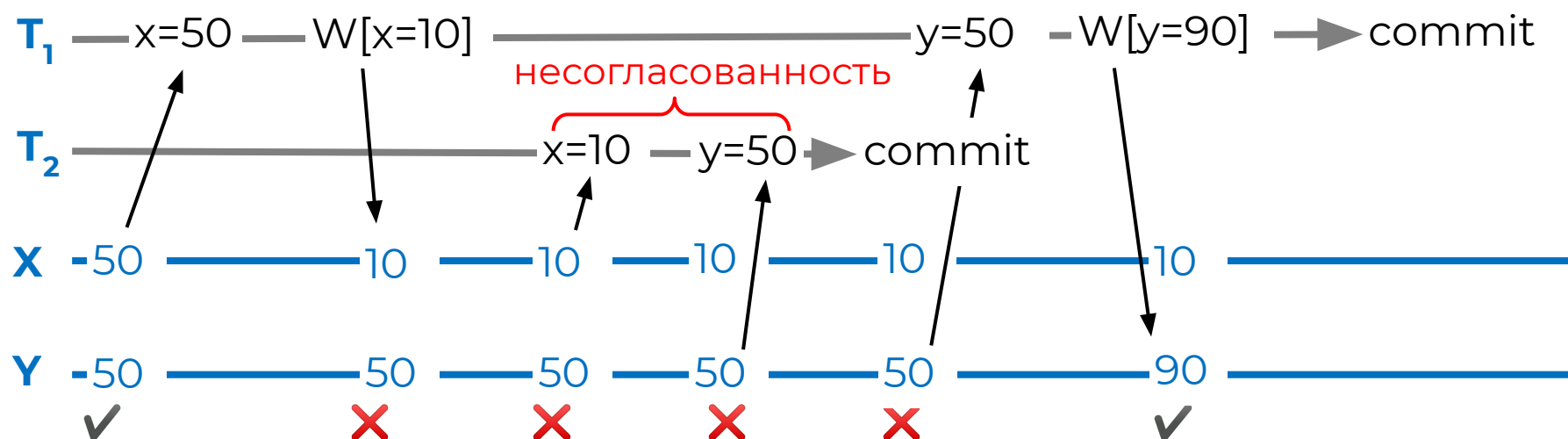
Используется по умолчанию для большинства баз данных.

Гарантии:

- транзакции не перезаписывают данные друг друга в процессе исполнения (no dirty writes)
- транзакции не могут прочитать данные, которые записывает другая транзакция (no dirty reads)

# Dirty Reads

Часть исполненной транзакции видна для других пользователей.



**Ограничение:  $x + y = 100$**

Другие транзакции могут основываться на отменённых данных.  
Нарушается причинно-следственная связь.

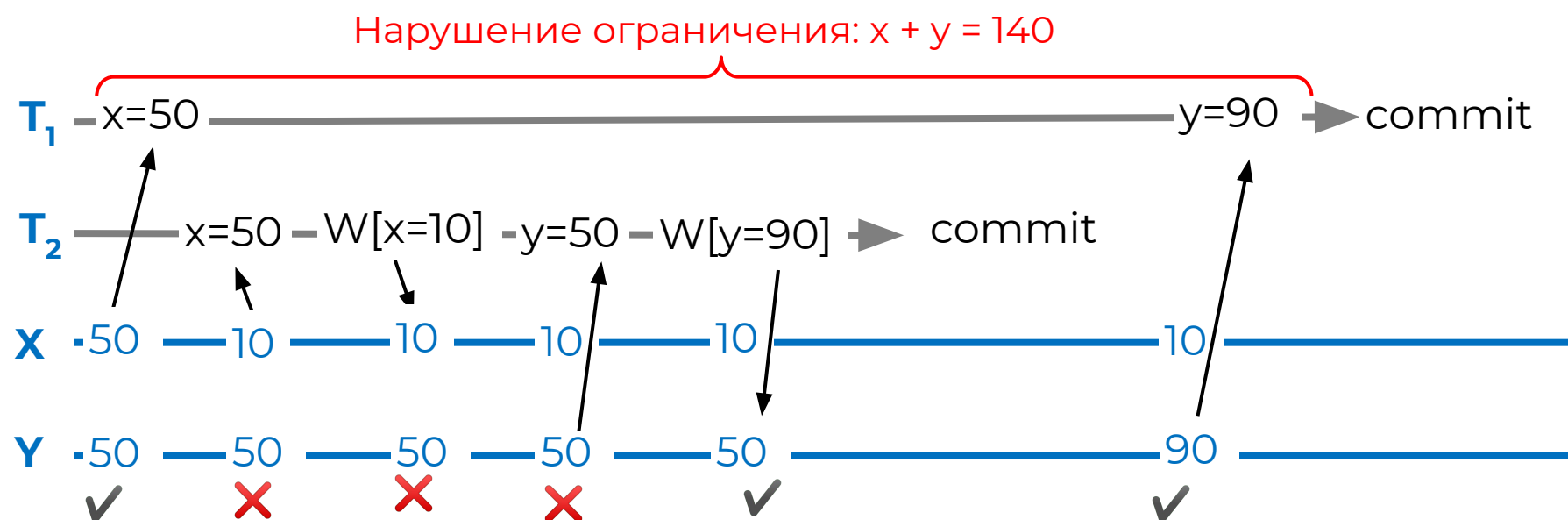
# Repeatable Read

Гарантии:

- транзакции не перезаписывают данные друг друга в процессе исполнения (no dirty writes)
- транзакции не могут прочитать данные, которые записывает другая транзакция (no dirty reads)
- нет потерянных обновлений (no lost updates)
- стабильность чтения данных (no fuzzy reads)
- нет смещений чтения/записи данных (no read/write skew)

# Fuzzy Read

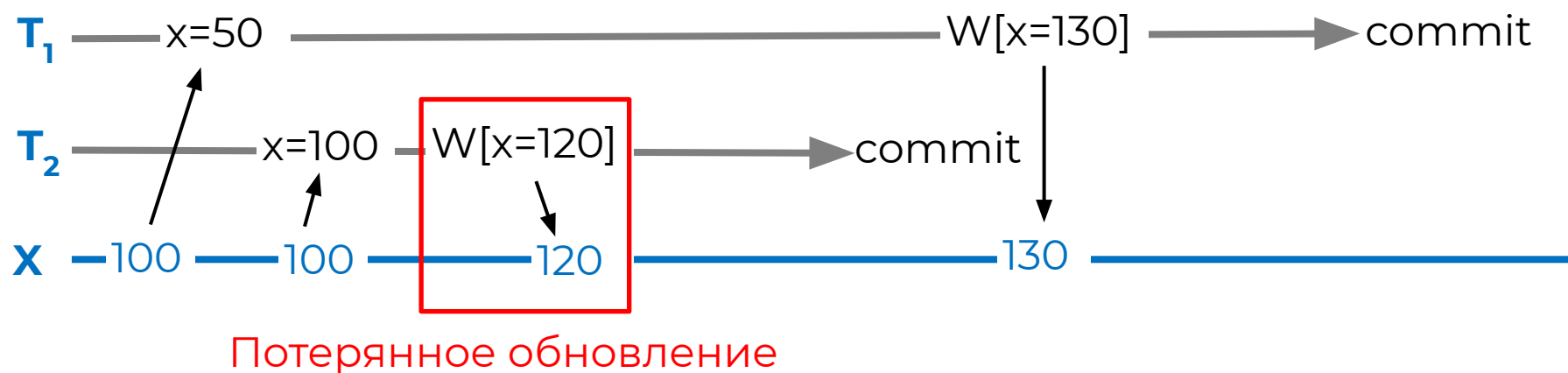
Результат чтения в рамках одной транзакции может меняться, и правила согласованности нарушаются для одной из транзакций.



Ограничение:  $x + y = 100$

# Lost Updates

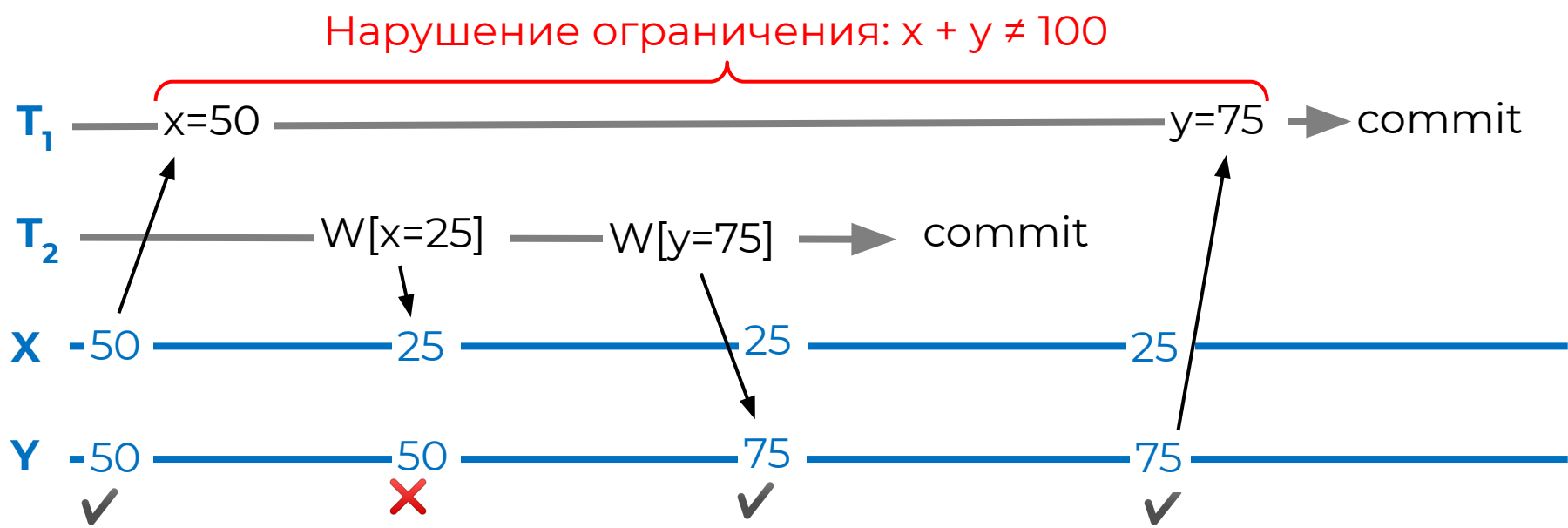
Транзакции одновременно обновляют одну и ту же запись.



Для решения проблемы БД ставит блокировку на все данные, которые вернулись в результате запроса.

# Road Skew

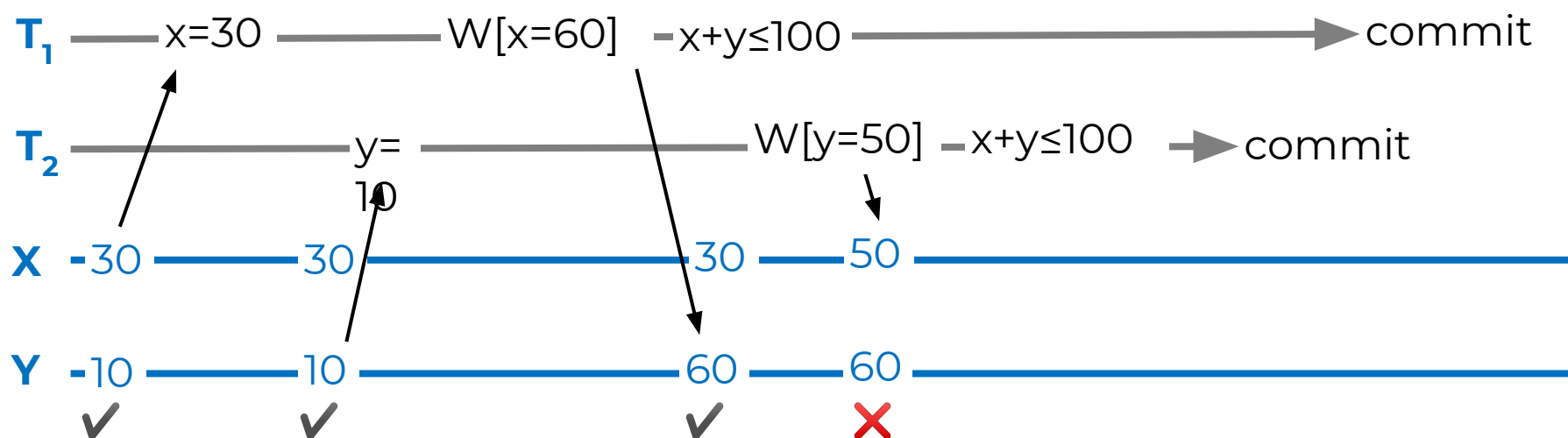
Возникает, когда нарушается внутреннее ограничение на согласованность данных.



Ограничение:  $x + y = 100$

# Write Skew

Для каждой из одновременно идущих транзакций ограничение не нарушается, но итоговый результат нарушает ограничения БД.



**Ограничение:  $x + y \leq 100$**

Обновляются разные записи, поэтому это не Dirty Write и не Lost Updates.

# Snapshot Isolation

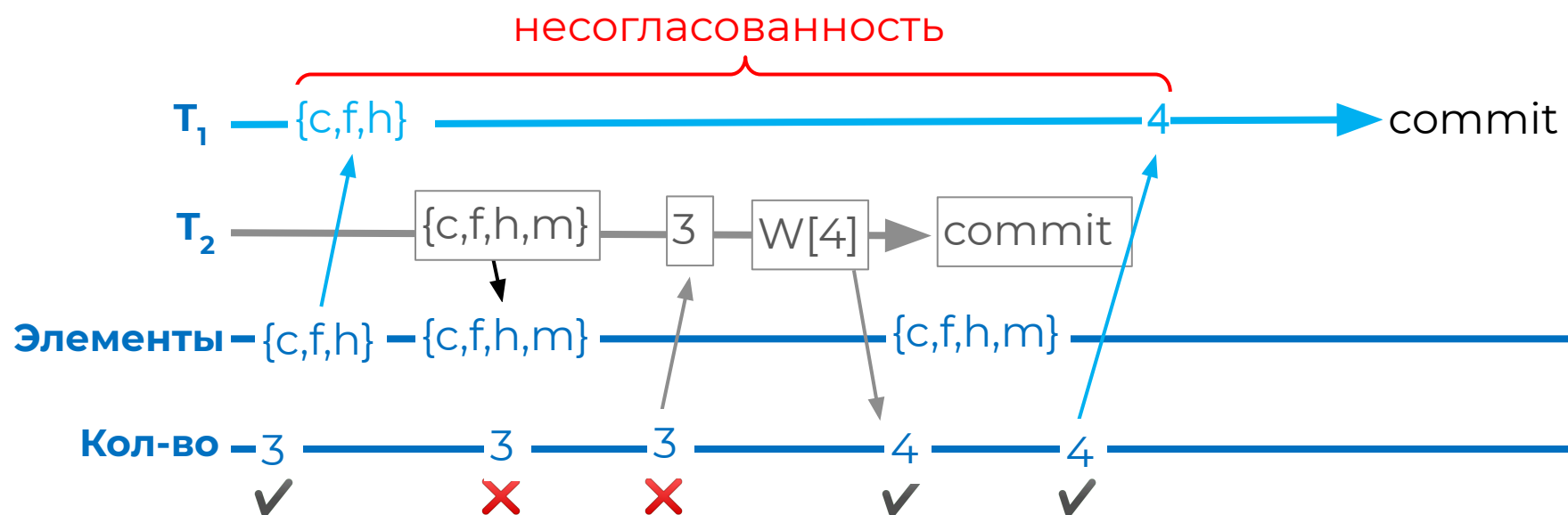
Гарантии:

- транзакции не перезаписывают данные друг друга в процессе исполнения (no dirty writes)
- транзакции не могут прочитать данные, которые записывает другая транзакция (no dirty reads)
- нет потерянных обновлений (no lost updates)
- есть стабильность чтения данных (no fuzzy reads)
- нет смещений чтения/записи данных (no read skew)
- нет фантомных данных во время записи (phantoms)



# Phantoms

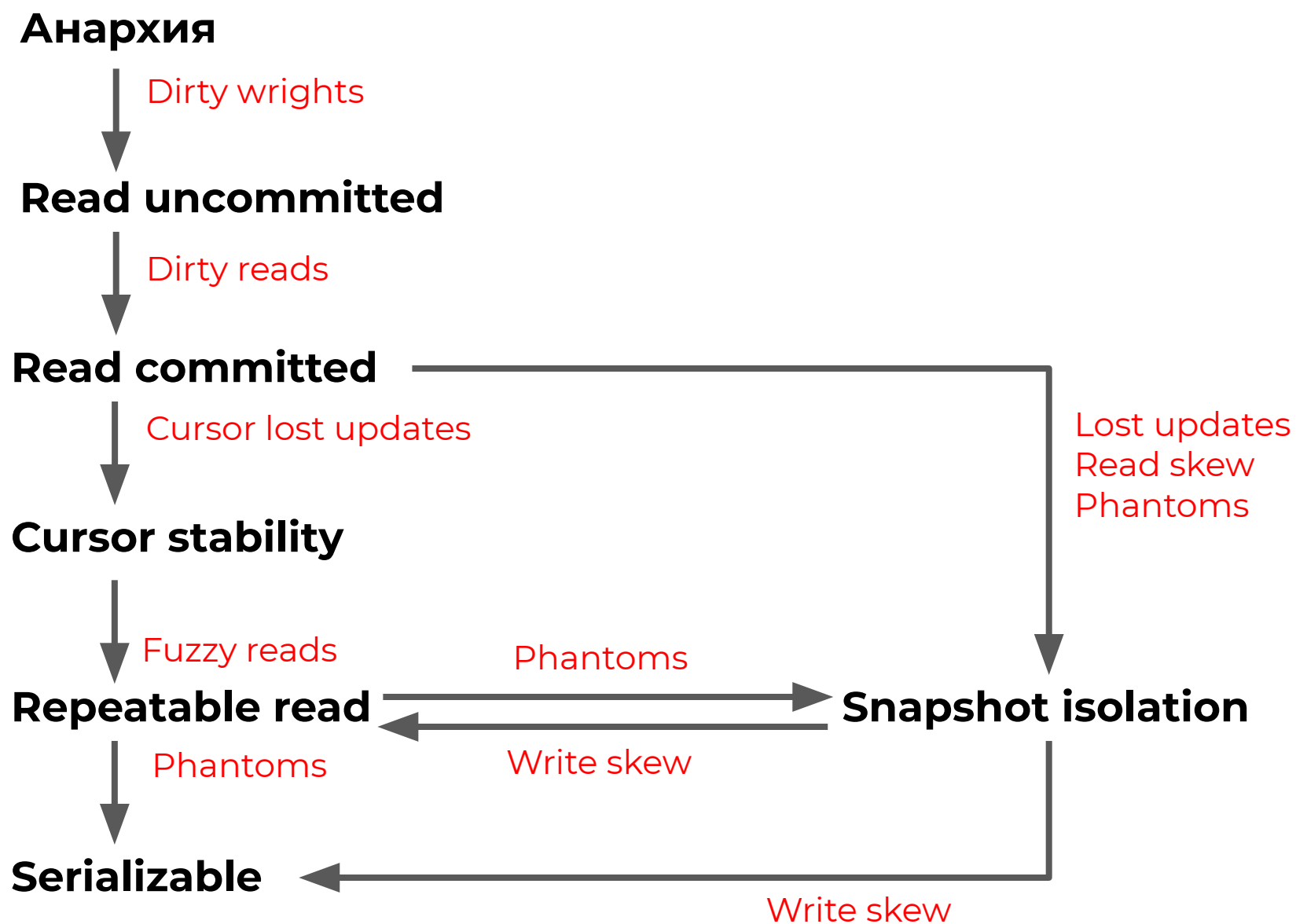
Фантомные записи возникают во время использования предиката для чтения, когда такой же предикат используется другой транзакцией для записи во время действия первой транзакции.



# Serializable

Гарантии:

- транзакции не перезаписывают данные друг друга в процессе исполнения (no dirty writes)
- транзакции не могут прочитать данные, которые записывает другая транзакция (no dirty reads)
- нет потерянных обновлений (no lost updates)
- есть стабильность чтения данных (no fuzzy reads)
- нет смещений чтения/записи данных (no read/write skew)
- нет фантомных данных во время записи (phantoms)



Чем выше уровень изоляции данных,  
тем медленнее общая пропускная  
способность базы данных

# Альтернатива

Если изменения данных приводят к аномалиям и несогласованности, то...



...надо данные только дописывать и никогда не менять.

# Append-only

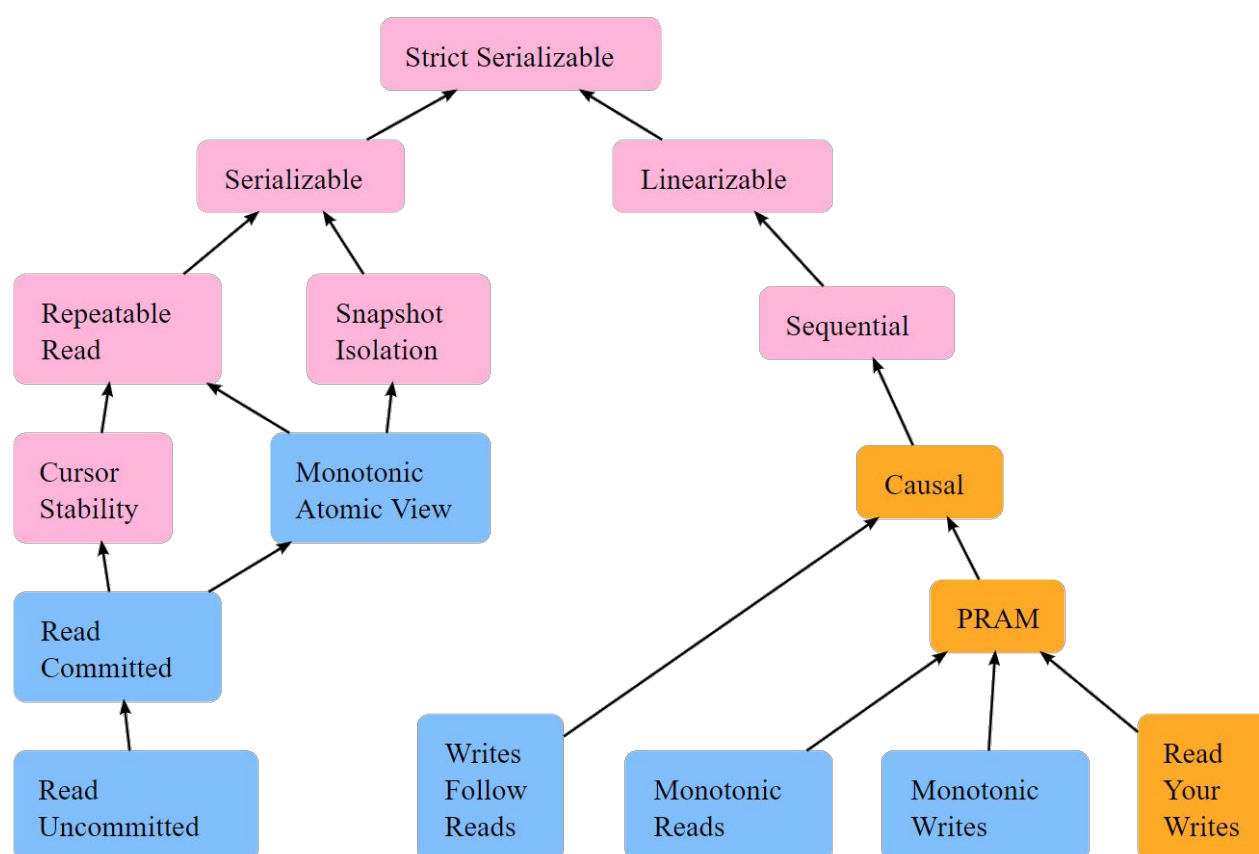
Вопросы для решения:

- сохранение причинно-следственных связей
- конфликты версий

Возможные решения:

- двухфазное подтверждение
- lamport timestamps
- paxos
- raft

# Модели согласованности



Данные недоступны во время некоторых сбоев сети. Все или несколько серверов БД должны приостановить свою работу для обеспечения надёжности данных.

Данные доступны на всех рабочих серверах, пока клиент общается с тем же самым сервером. При переключении данные недоступны.

Данные доступны на всех рабочих серверах, даже если сеть между ними полностью недоступна.

# Выводы

- Уровень изоляции влияет на производительность и возможные аномалии данных — необходимо работать с компромиссами.
- Необходимо изучать и знать принципиальные возможности движков баз данных, чтобы принимать решения, отвечающие ФТ и НФТ.



# Итоги модуля

- Транзакция — логическая структура, помогающая в управлении согласованностью данных. Можно организовывать транзакции на уровне приложения или БД — решение за вами.
- Распределённые транзакции имеют сильные штрафы к производительности и риски; использовать стоит, когда нет другого выхода и согласованность данных превыше скорости.
- Если необходимо использовать «Сагу», то используйте с большой осторожностью. Посмотрите ещё раз на альтернативы.
- Модель согласованности данных должна быть определена исходя из бизнес-требований и ограничений.

Skillbox

**Спасибо  
за внимание!**