

Skillbox

Работа с событиями. Event Sourcing

Андрей Гордиенков

Solution Architect

ABAX

В прошлом уроке

- Концепт события
- Типы событий и их применение

В этом уроке

- Какие проблемы решает и создаёт Event Sourcing?
- Технические нюансы реализации

«Традиционные» проблемы

Состояние объекта:

ID	Name	SN	Price	Unit	Blueprint	Description	Dimensions	Type
1544845323	Veronika	T045-45884	35.50	23	332241	Lorem ipsum	12x3x44	Table

«Традиционные» проблемы

Состояние объекта через секунду:

ID	Name	SN	Price	Unit	Blueprint	Description	Dimensions	Type
1544845323	Veronika	T045-46884	39.50	23	332241	Lorem ipsum	12x3x45	Table

чѐт падазритильна



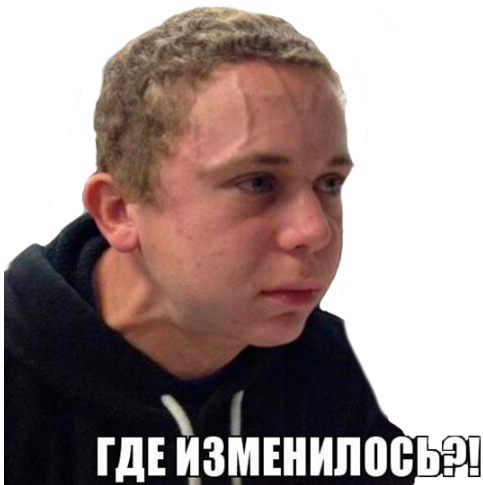
«Традиционные» проблемы

Было

ID	Name	SN	Price	Unit	Blueprint	Description	Dimensions	Type
1544845323	Veronika	T045-45884	35.50	23	332241	Lorem ipsum	12x3x44	Table

Стало

ID	Name	SN	Price	Unit	Blueprint	Description	Dimensions	Type
1544845323	Veronika	T045-46884	39.50	23	332241	Lorem ipsum	12x3x45	Table



«Традиционные» проблемы

И даже если разобрались, что изменилось:

ID	Name	SN	Price	Unit	Blueprint	Description	Dimensions	Type
154484 5323	Veronika	T045-4 6884	39.50	23	332241	Lorem ipsum	12x3x45	Table

Почему?



Хранение состояния

Плюсы:

- просто
- знакомо

Минусы:

- причина изменения непонятна
- жёсткая связь «клиент — хранилище — формат»
- невозпроизводимость для отладки
- необходимость «вытягивания» данных

Event Sourcing предлагает решение этих проблем.

Event Sourcing



“
Моделирование информации об активности в домене в виде обособленных событий. Каждое событие также является доменным объектом.

Eric Evans, Domain-Driven Design Reference

События и команды

Команда — запрос в «будущее», надежда, что оно наступит. Название всегда в повелительном наклонении.

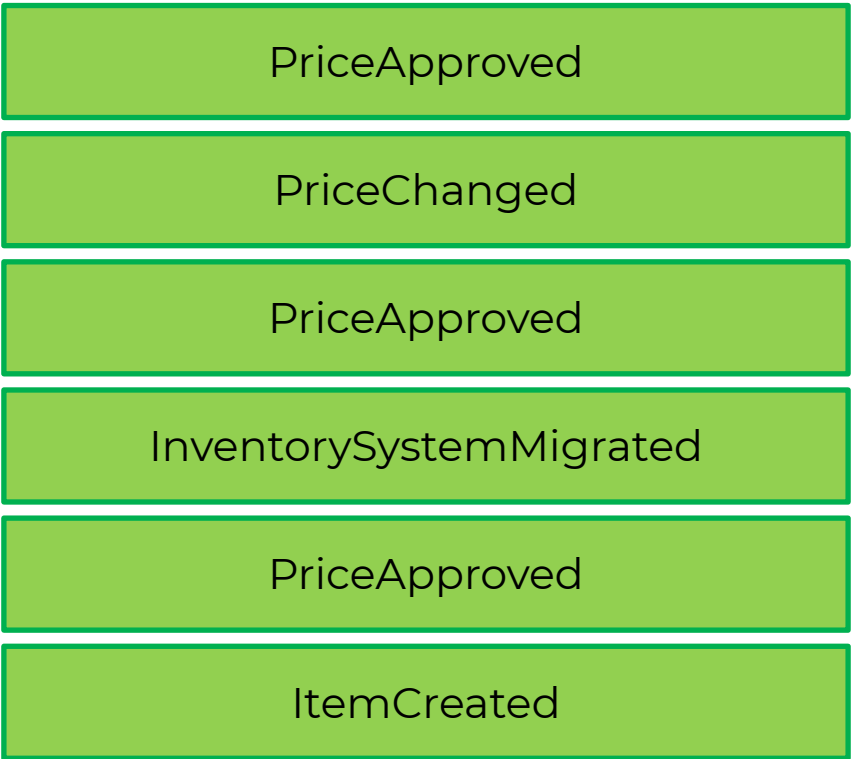
Событие — сообщение о том, что «будущее» наступило, что-то где-то произошло. Событие не отменить, на него можно только реагировать или не реагировать. Название всегда в прошедшем времени.

Event Sourcing опирается на события, потому что они представляют факты.

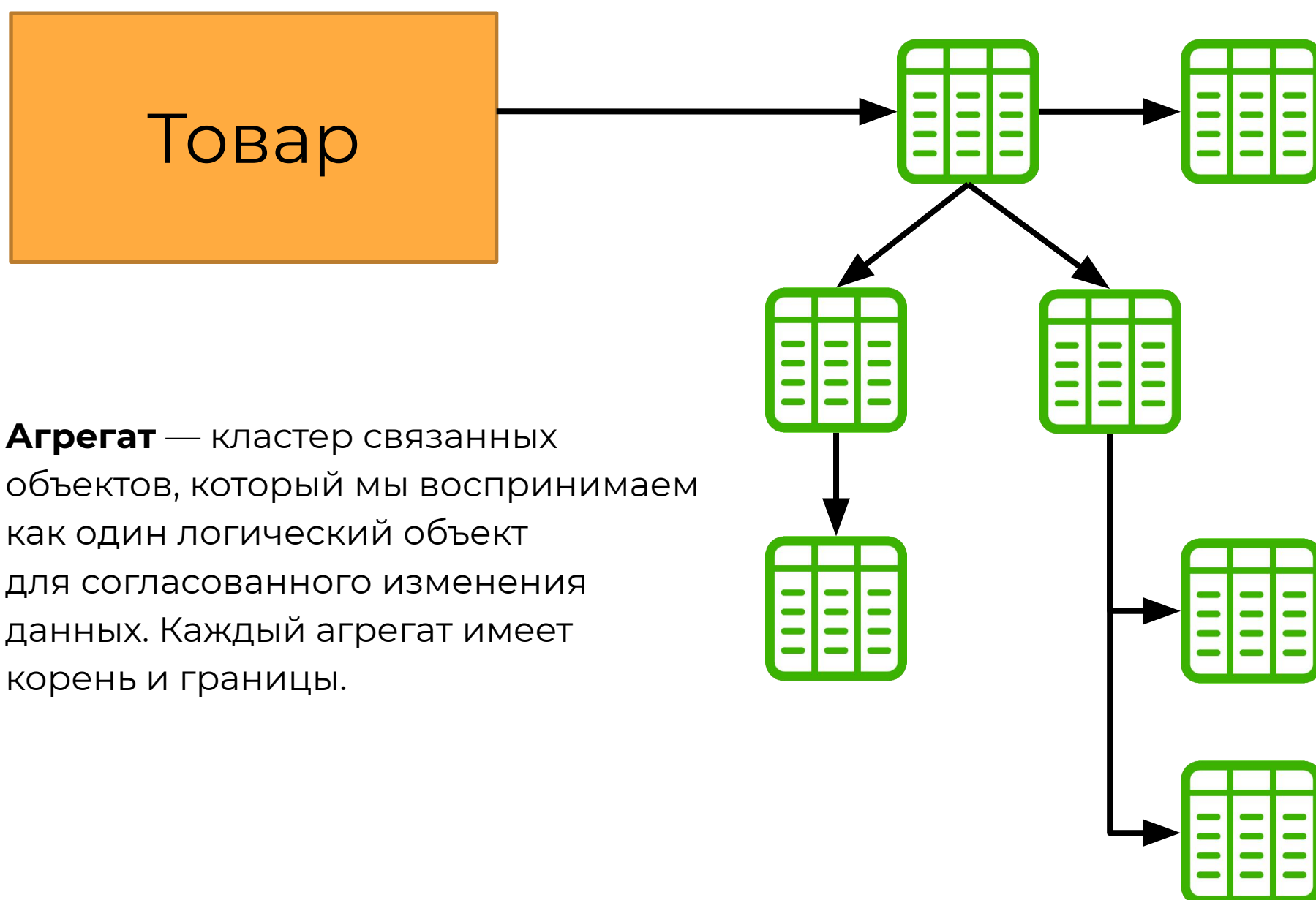
Возможная история

Что привело к данному состоянию:

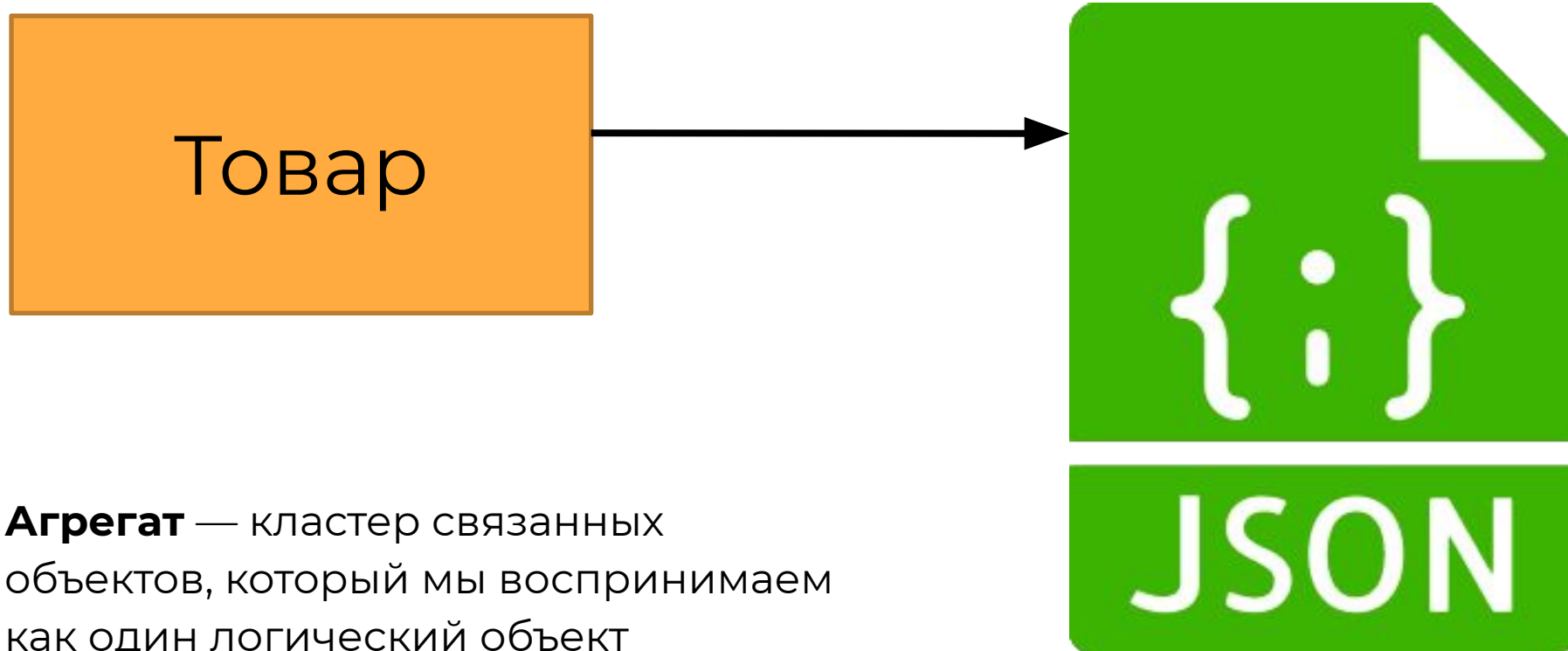
ID	Name	SN	Price	Unit	Blueprint	Description	Dimensions	Type
1544845323	Veronika	T045-46884	39.50	23	332241	Lorem ipsum	12x3x45	Table



Агрегаты



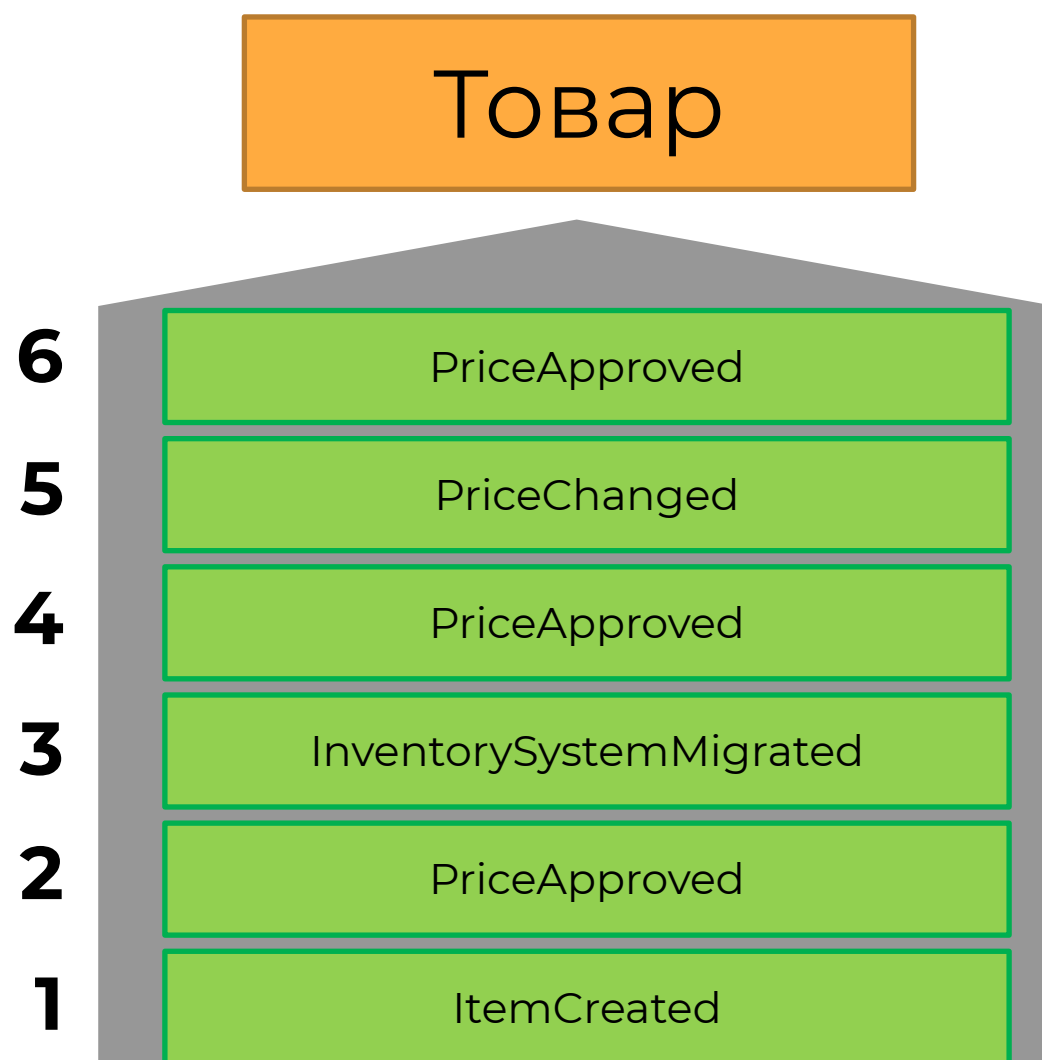
Агрегаты



Агрегат — кластер связанных объектов, который мы воспринимаем как один логический объект для согласованного изменения данных. Каждый агрегат имеет корень и границы.

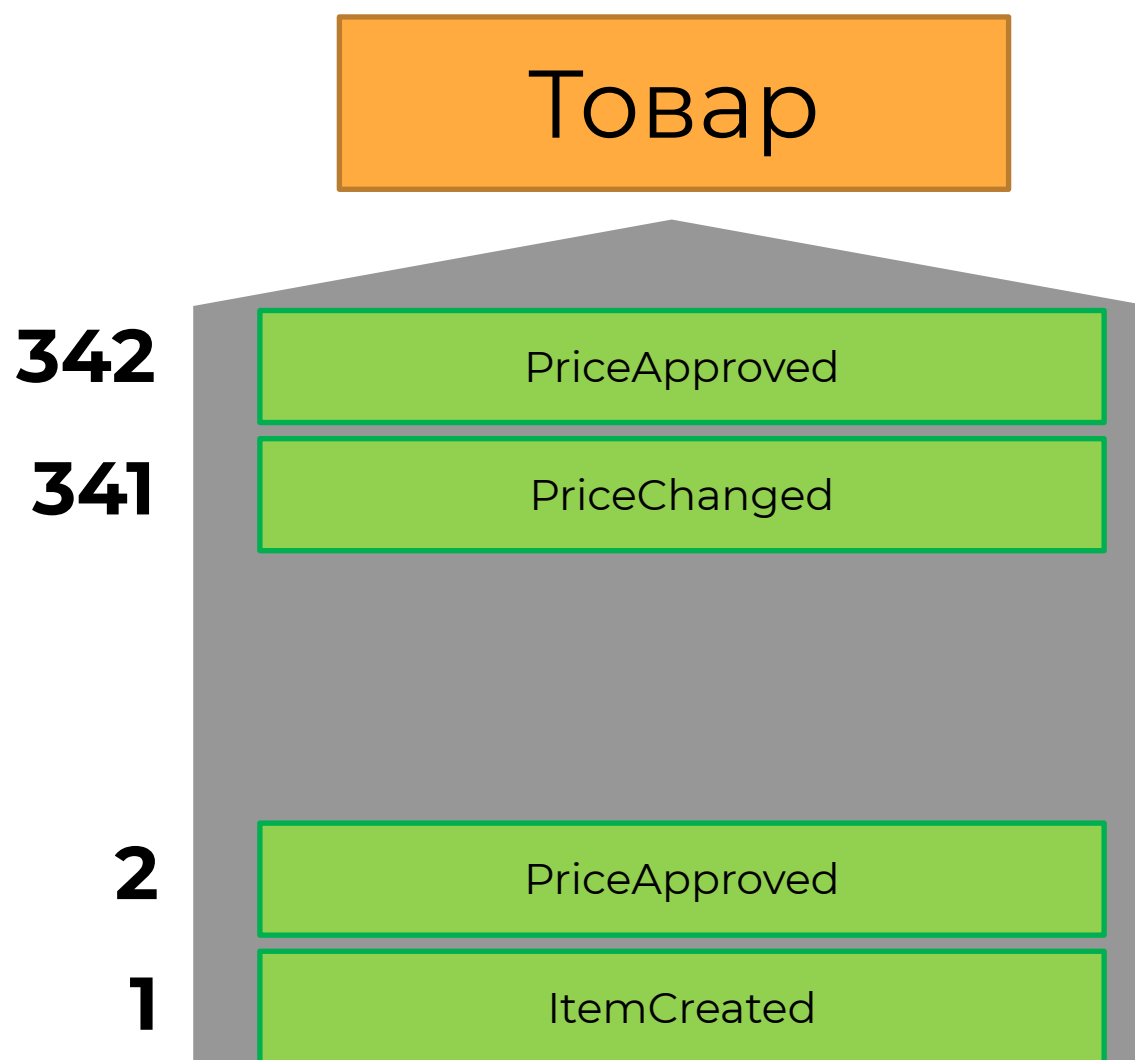
Восстановление данных

Восстановление идёт от самого первого события.

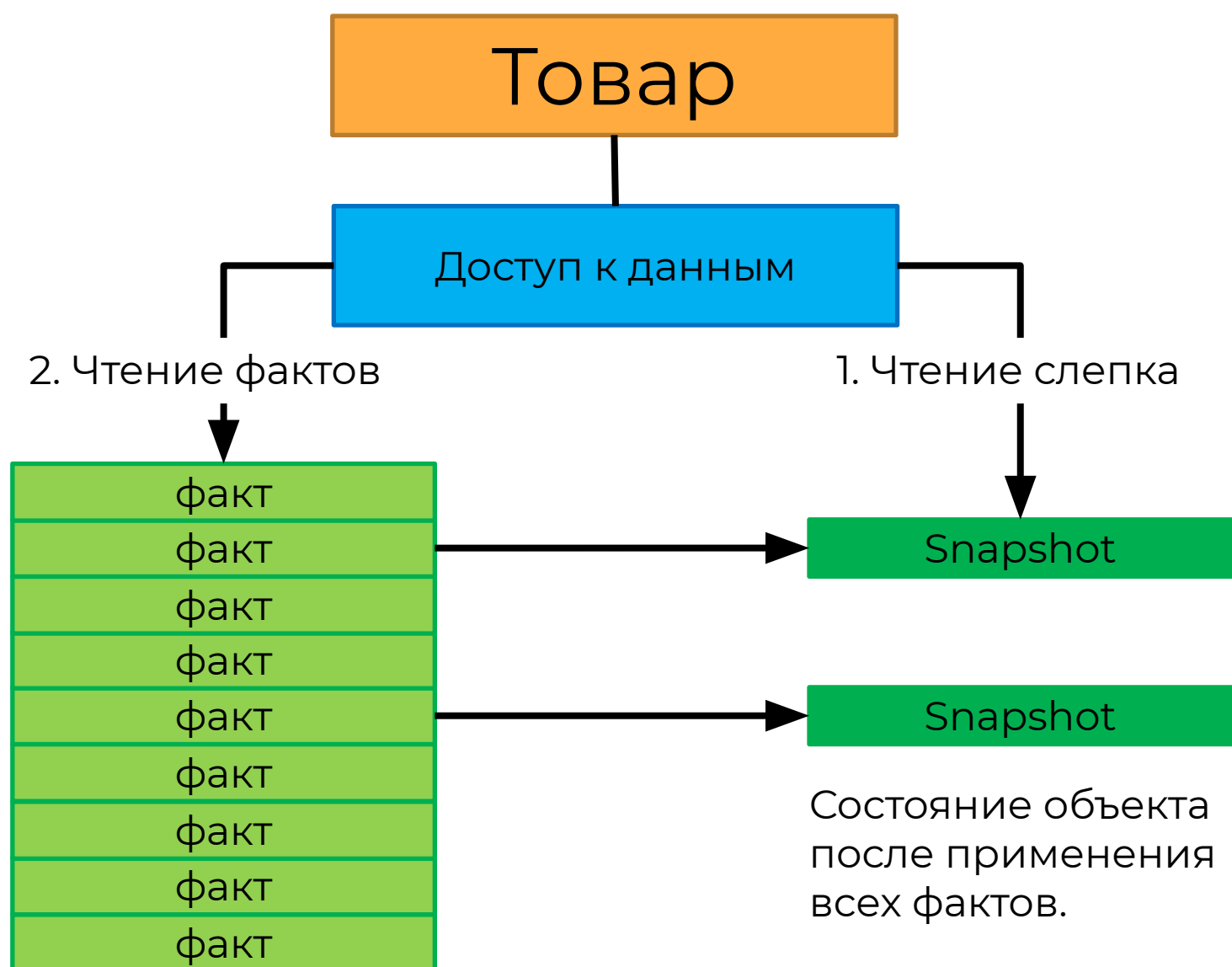


Восстановление данных

Если фактов изменения очень много, то что делать?

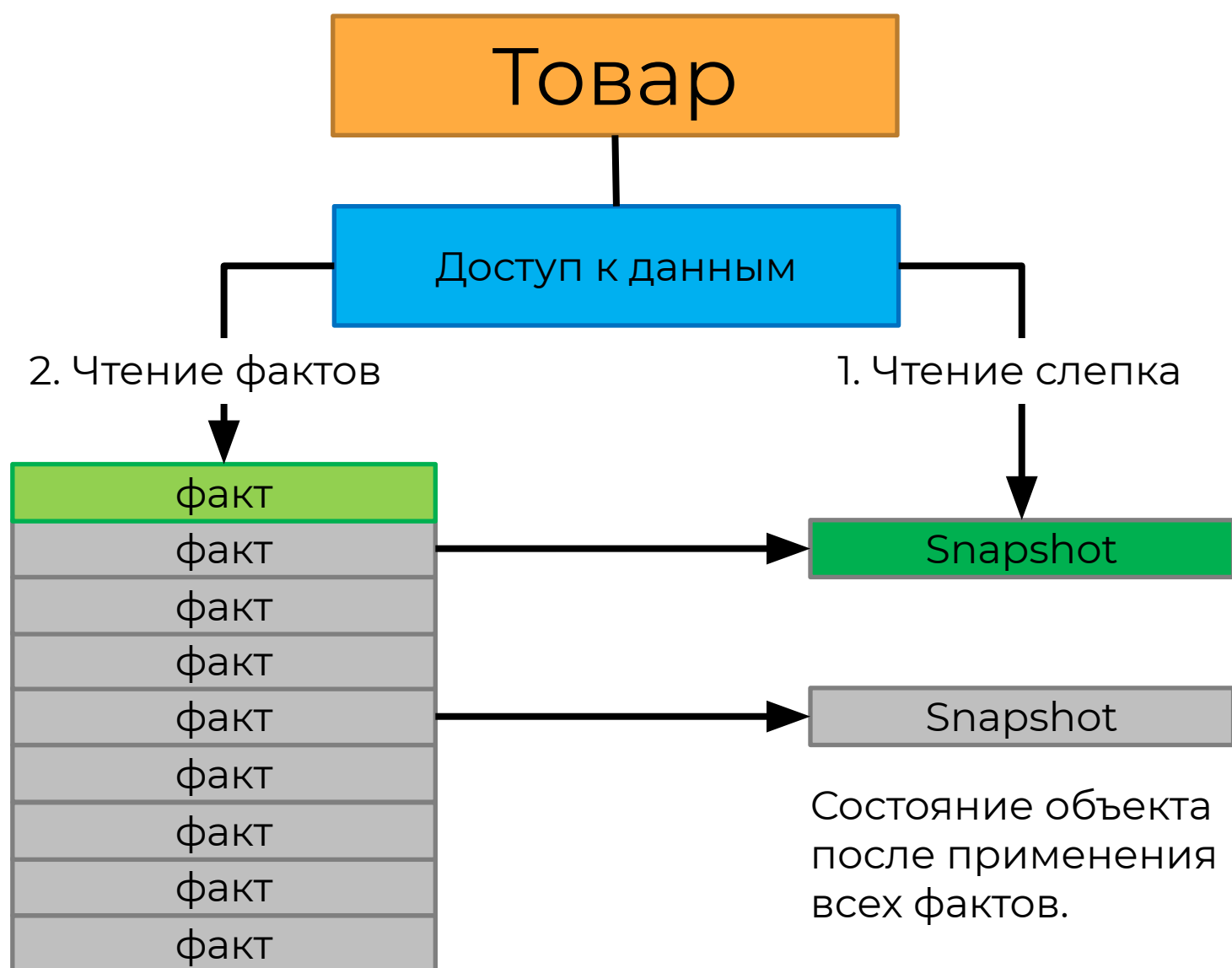


Snapshots

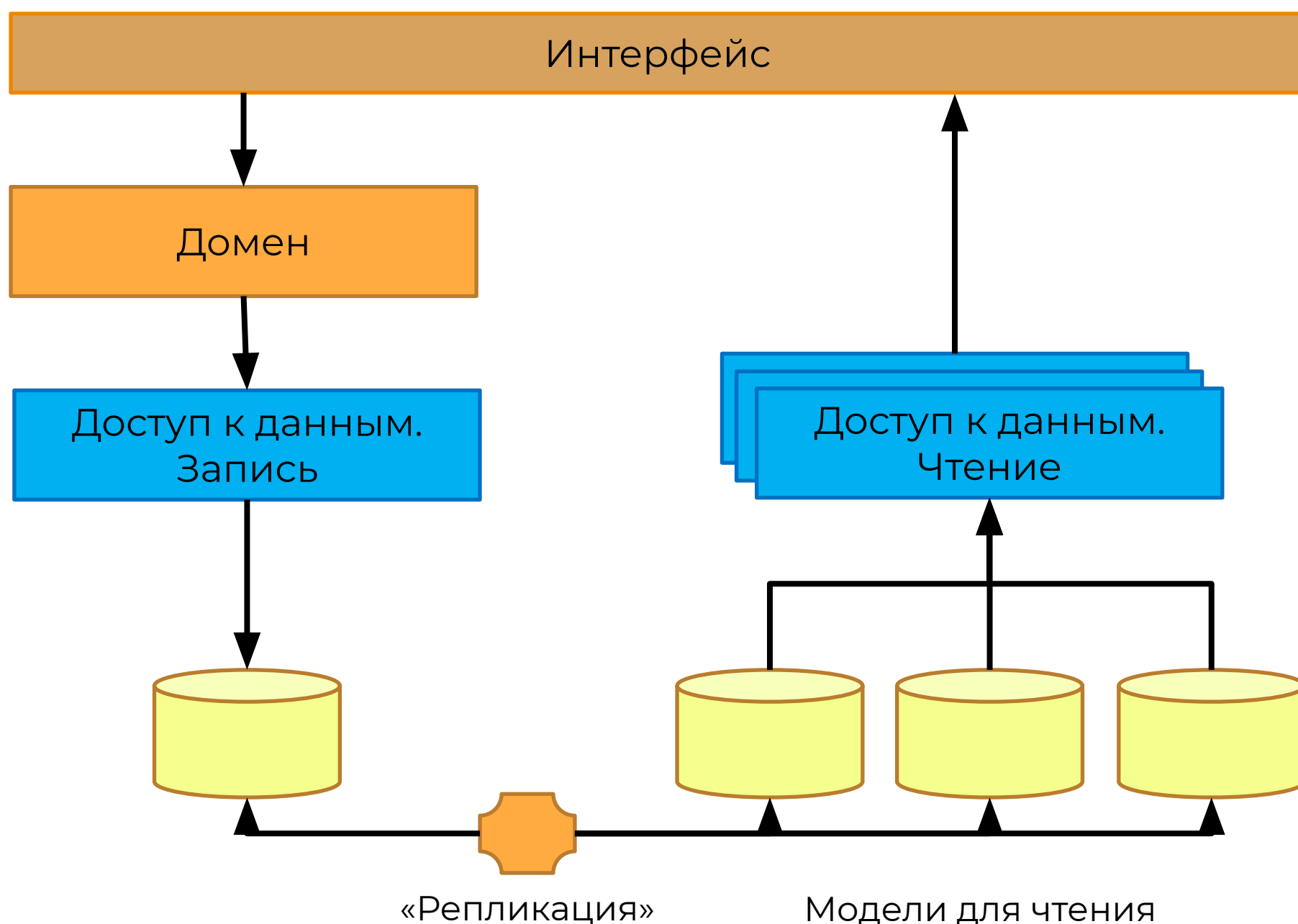


Snapshots

Слепки данных ускоряют получение финального состояния.



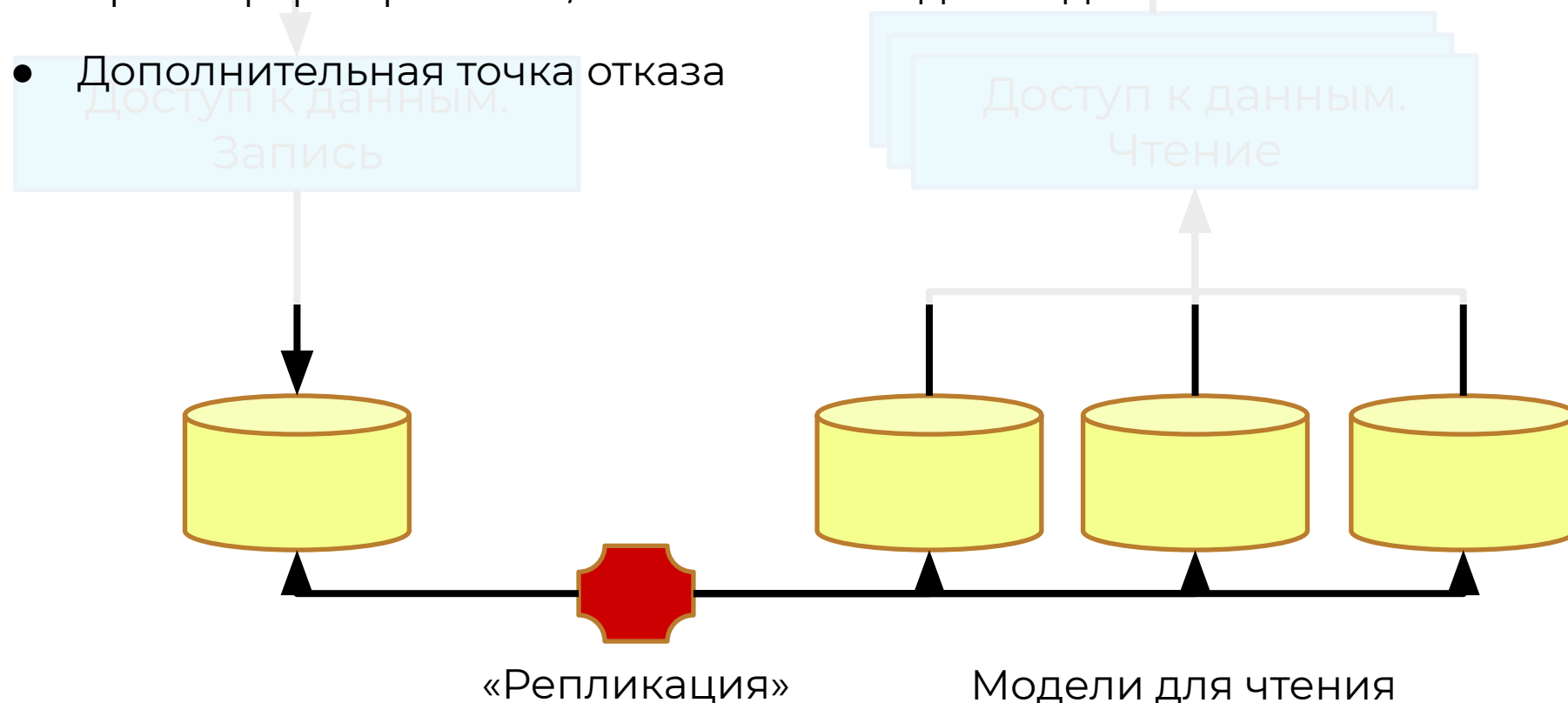
Модель для чтения



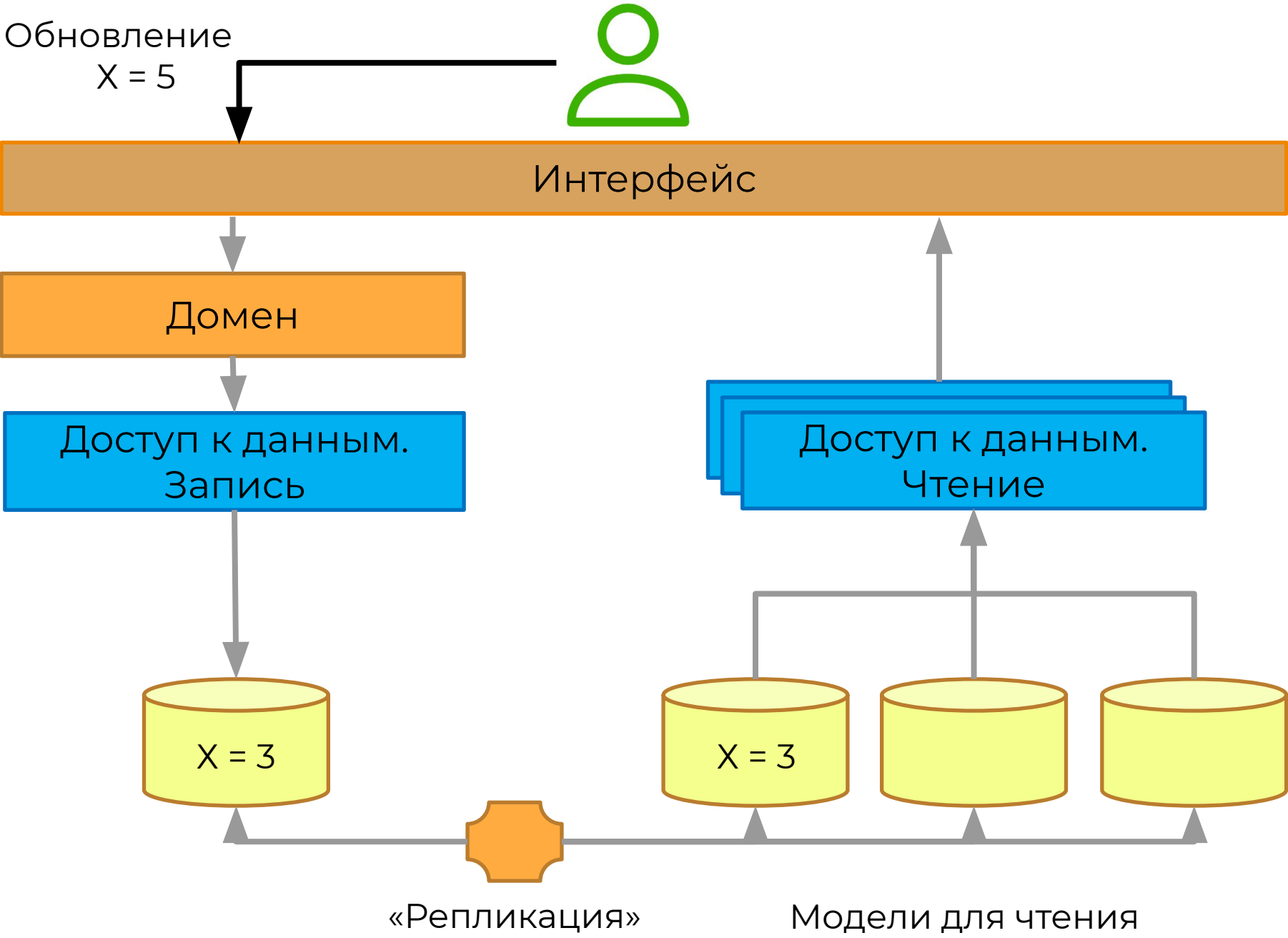
«Репликация»

Одна из самых проблемных частей всего подхода.

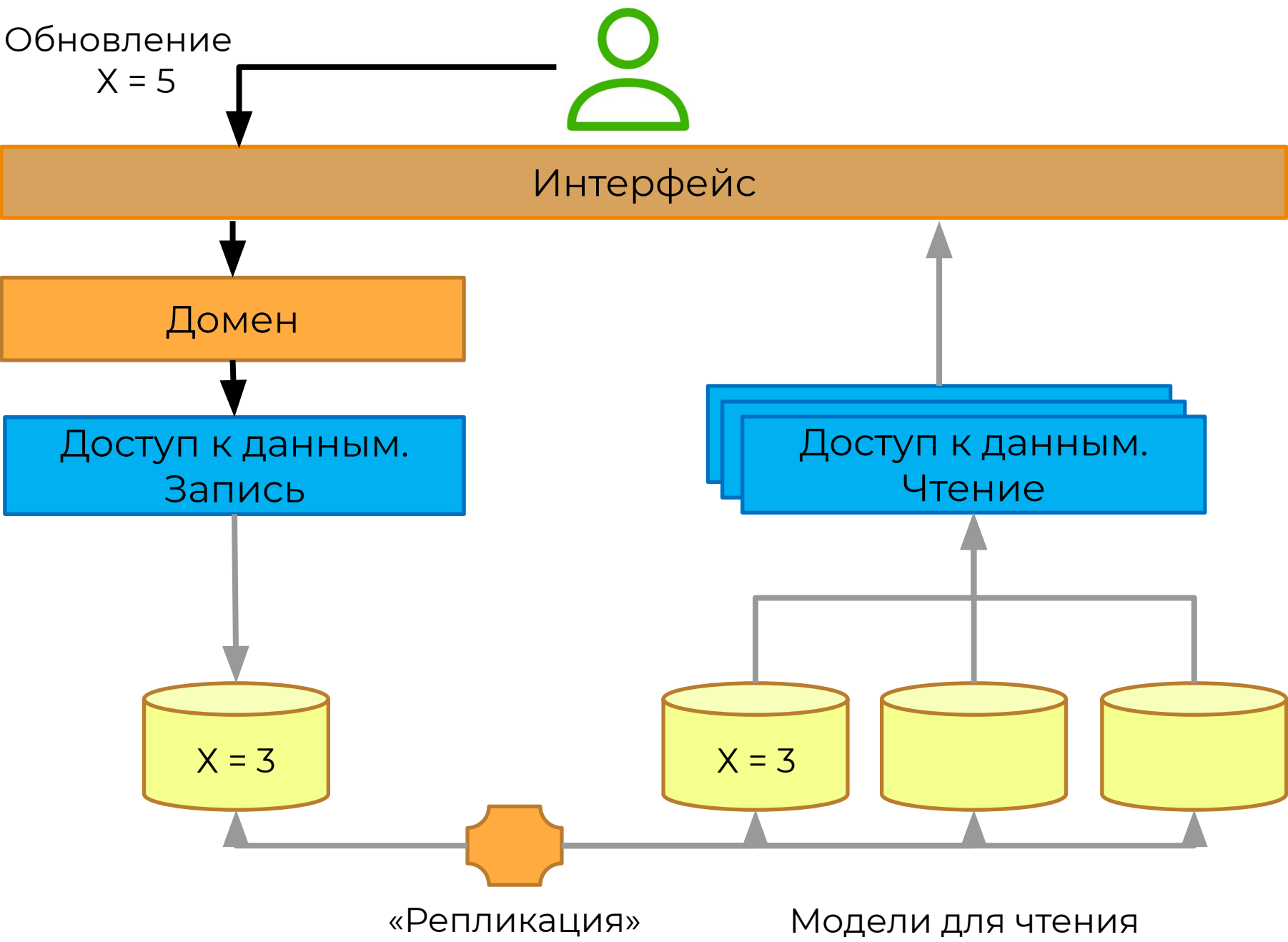
- Дополнительная логика для копирования данных в формат для чтения
- Сильная избыточность данных
- Время формирования/обновления моделей для чтения
- Дополнительная точка отказа



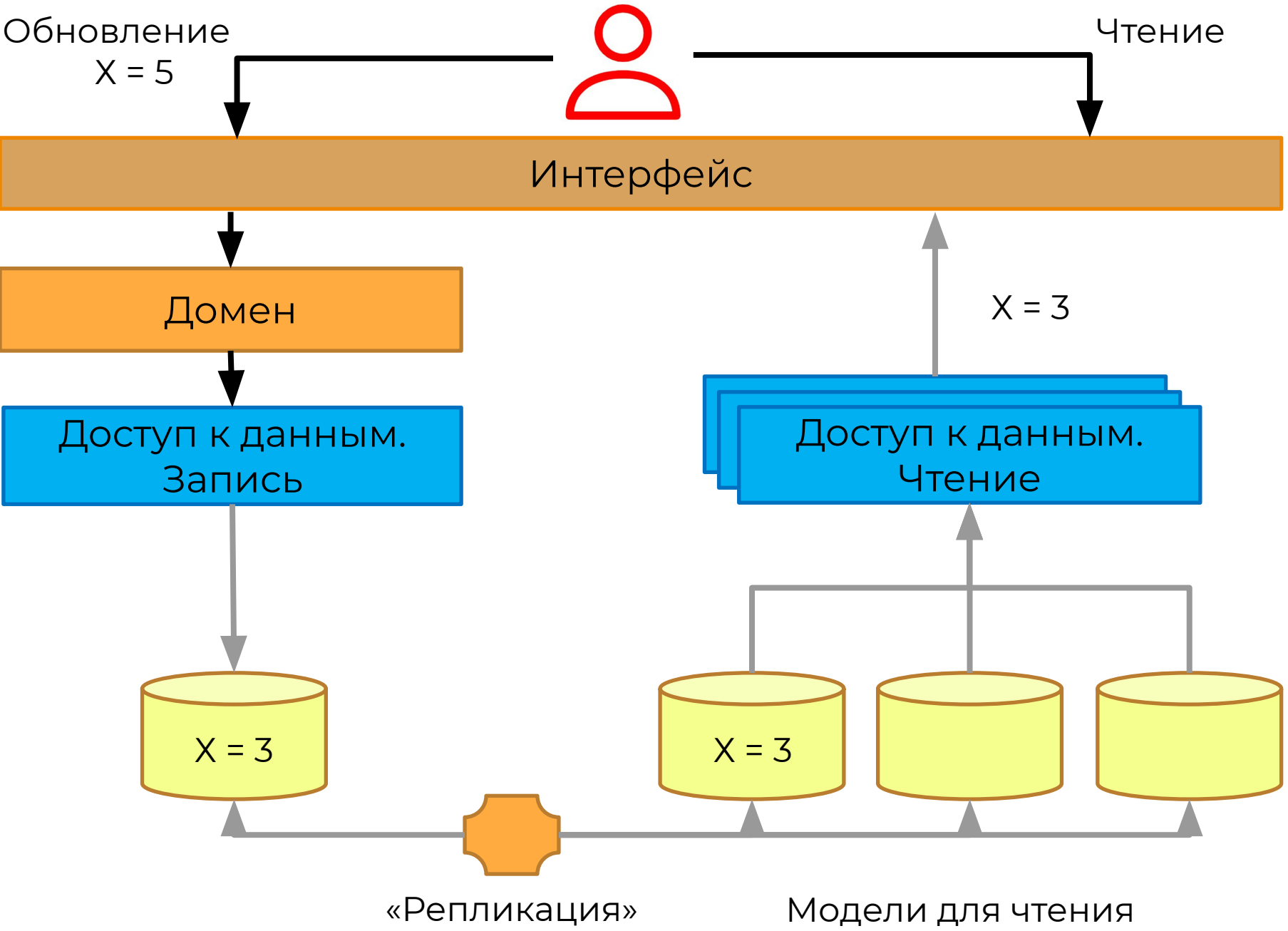
Модель для чтения



Модель для чтения



Модель для чтения



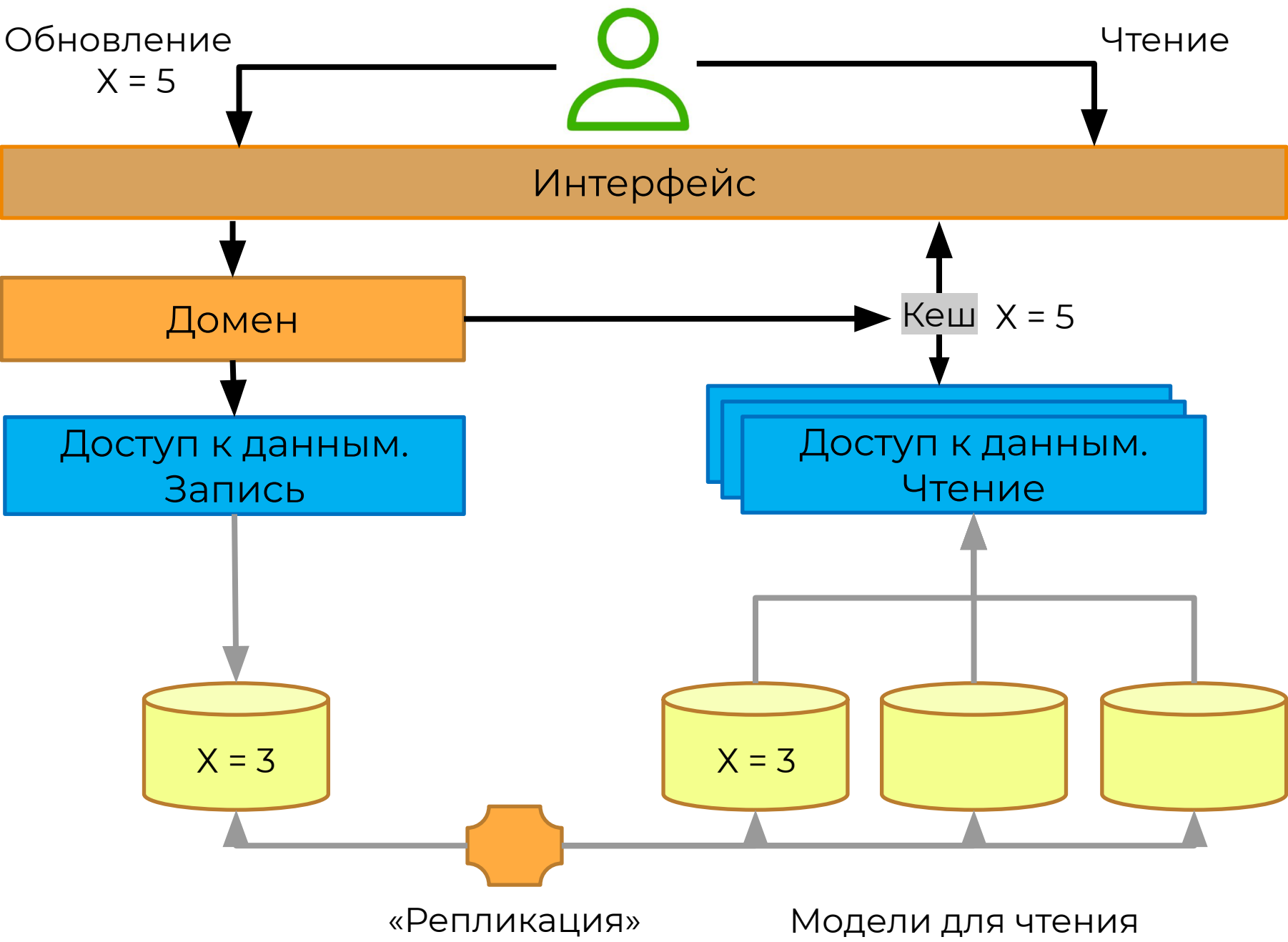
Чтение своих данных

Может возникнуть ситуация, когда данные для записи не успели обновиться в моделях для чтения.

Что делать?

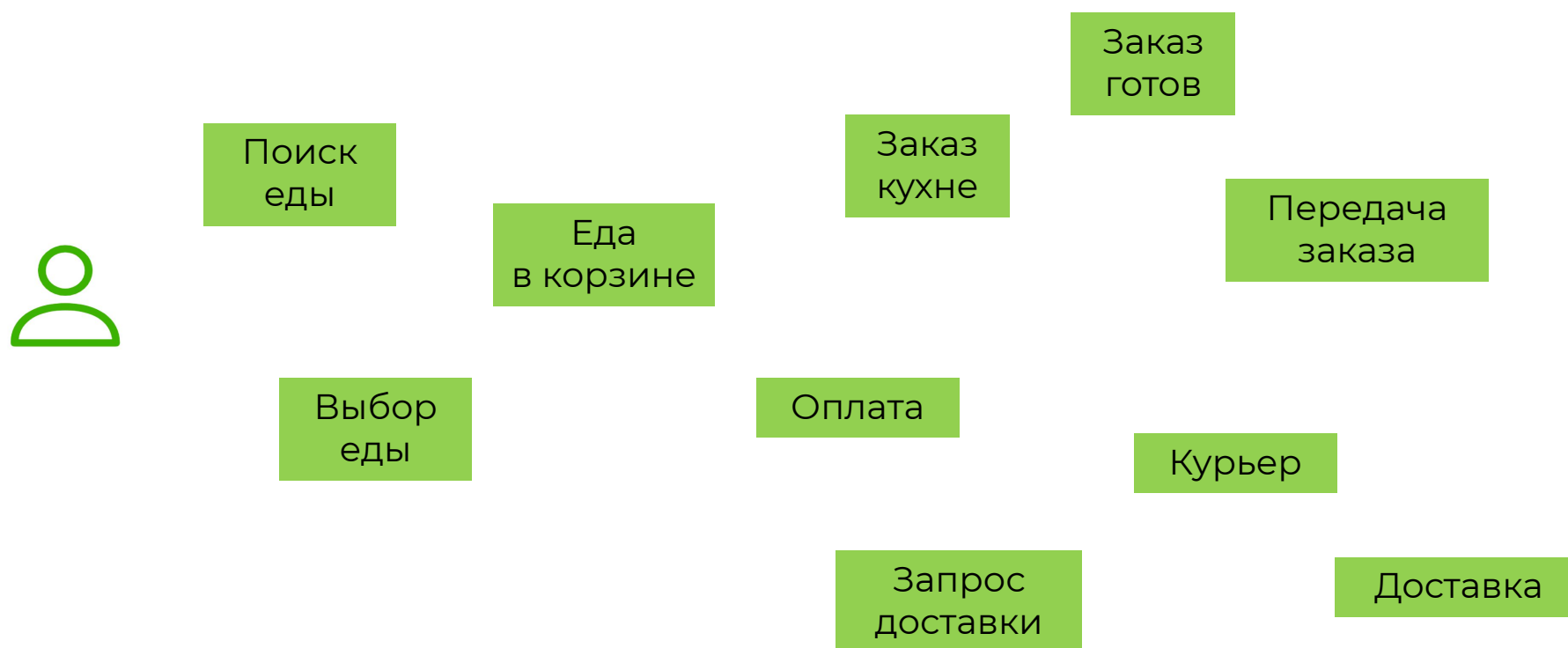
Делать кратковременный кеш данных для пользователя.

Чтение своих данных



Откуда взять события?

Для выявления и моделирования событий используется техника Event Storming.



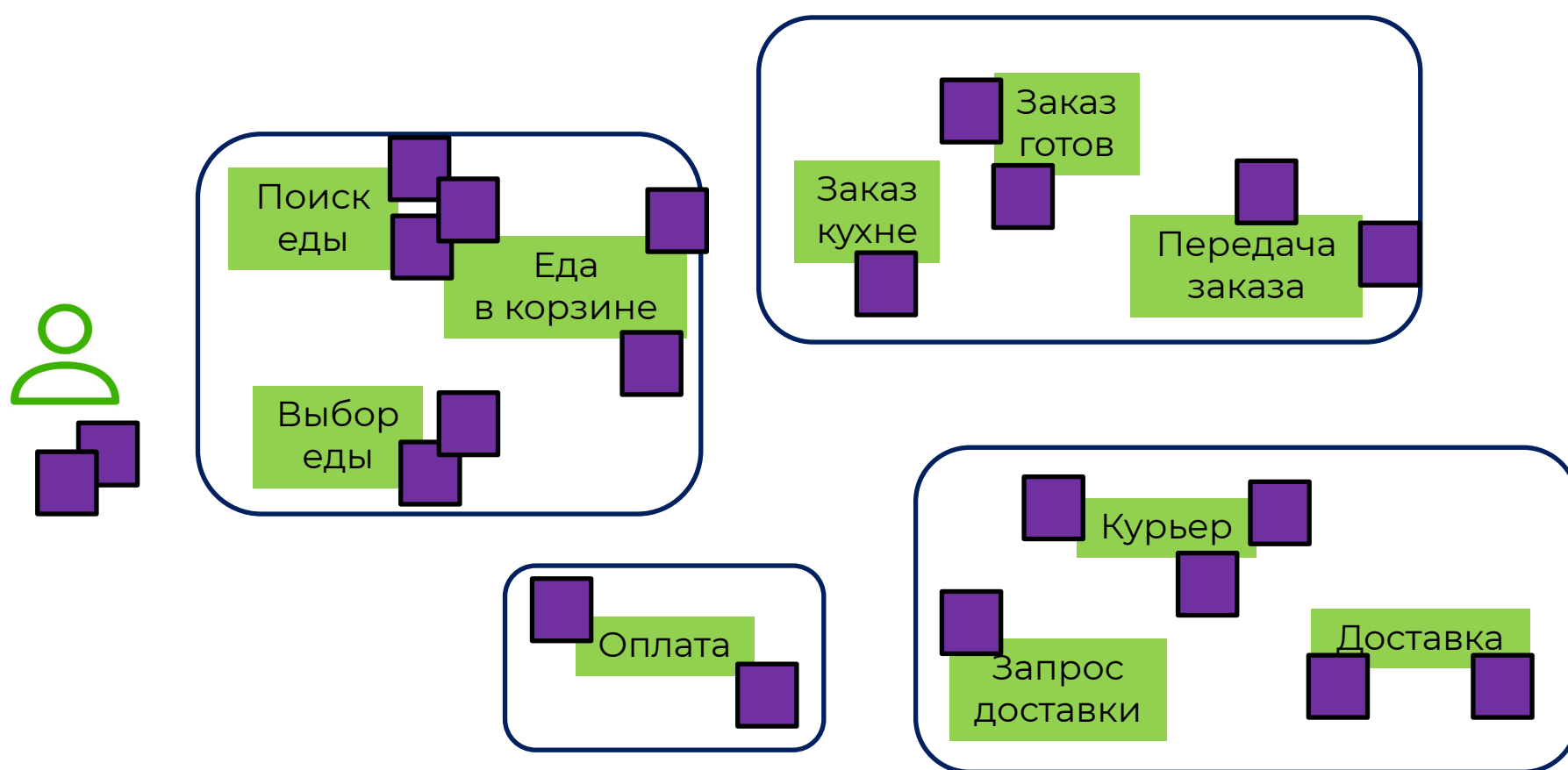
Откуда взять события?

Для выявления и моделирования событий используется техника Event Storming.



Откуда взять события?

Для выявления и моделирования событий используется техника Event Storming.



Event Storming

- Для проведения требуются доменные эксперты
- Определяются ключевые события для основных бизнес-процессов
- Не забыть о различного рода отменах
- Осторожно подходить к границам сервисов. Желательно, чтобы в рамках одного сервиса был только один агрегат
- Основной способ общения сервисов — реакция на факты, а не на прямые запросы что-то сделать

Программное обеспечение

Для хранения событий подойдёт любое хранилище, которое не диктует жёсткую схему хранения данных.

Есть специализированная база Event Store.

Фреймворки для Event Sourcing не нужны.

События — обычные классы или структуры.

Тестирование

Проверяется наличие событий и их состав.

- Легко проверить на отсутствие сайд-эффектов
- Понятный импульс изменений

Плюсы Event Sourcing

- Хорошо отображает бизнес-процессы
- Явное описание причин изменений
- Возможность воспроизведения событий для отладки
 - Даже если сервис уже работает с новой версией сообщений
- Легче правильно определить границы сервисов с помощью Event Storming
- Система изначально готова к распределённой работе, потому что события неизменяемые и легко сериализуются
- Слабая связанность всей системы
- Готовый источник данных для аналитических систем

Минусы Event Sourcing

- Слабая связанность всей системы. Дополнительные усилия для построения зависимостей классов и событий
- Требуется больше места для хранения данных
- Восстановление данных сложнее
- Синхронизация лога событий с моделями для чтения
 - Сам механизм синхронизации
 - Проблема чтения своих данных
 - Поддержка версионирования
- Необходима предварительная работа по моделированию событий. Нужны хорошие бизнес-аналитики

Область применения

В основном для средних и крупных систем, где необходимы упор на гибкость решения и аналитические возможности.

Выводы

- Event Sourcing — мощный инструмент для построения сложных систем с аналитическими возможностями
- Высокая изменяемость решения и слабая связанность компонентов заложены в самой идее решения
- Высокая техническая сложность решения со многими дополнительными техническими компонентами, которые добавляют точки отказа
- Необходим опыт в построении доменных моделей

Что дальше?

- Command and Query Responsibility Segregation — шаблон и области его применения
- Связь с Event Sourcing

Skillbox

**Спасибо
за внимание!**