авиасейлс

Разнообразие требований к Data Warehouse

Артем Дайнеко

- Давление требований
- Daily метрики
- Timeliness
- Streaming Ingest
- Big Joins
- Joins. Интерлюдия
- Глубина дат
- Entity State
- High Cardinality
- Наивное решение High Cardinality. Интерлюдия
- Квалификация пользователей
- Пример

Давление требований

Ford Focus: https://unsplash.com/photos/W-982RIziS0
Formula One: https://unsplash.com/photos/QV0AinhRU2c

Решение по умолчанию

- Свежесть данных — 1 день
- Партиции по дням (или аналогичная сортировка)
- Загрузка через синхронизацию с RDBMS

1A Daily метрики

Всё как обычно, но расслабляться ещё рано

2 Timeliness

- Давление требований может быть очень велико, вплоть до самописных систем хранения данных
- Важны не столько пожелания, сколько процесс принятия решений: необходимый объём данных и скорость

3 Streaming Ingest

- Данные поступают быстро
- Но надо работать с дублями
- End to end principle

4 Big Joins

- Основной источник нагрузки для систем
  с невысокими требованиями к свежести данных
- Map-side join не доставляет проблем

Join. Интерлюдия

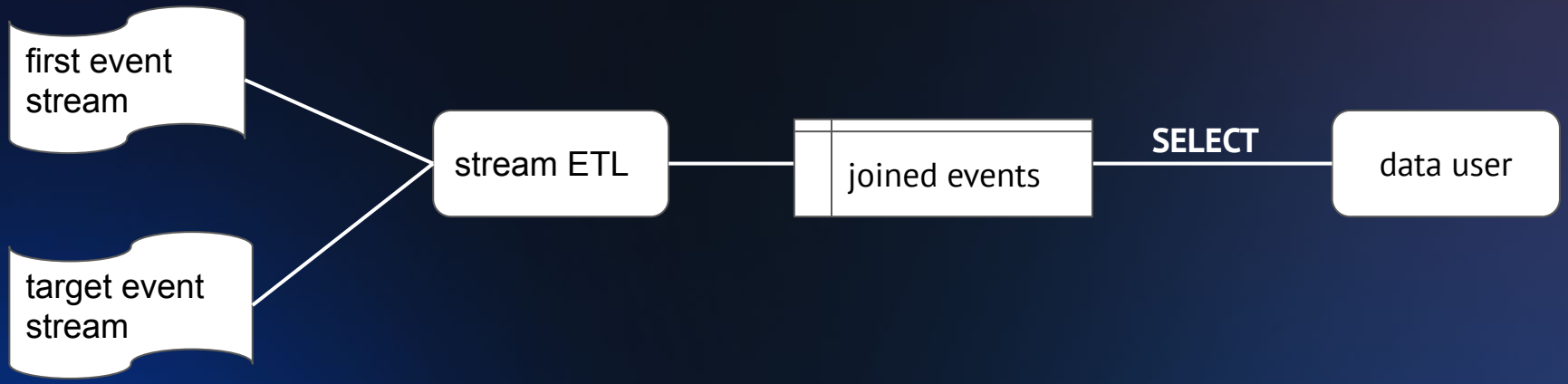
product → stream → aggregate → **runtime**

Join. Интерлюдия

product → stream → **aggregate** → runtime

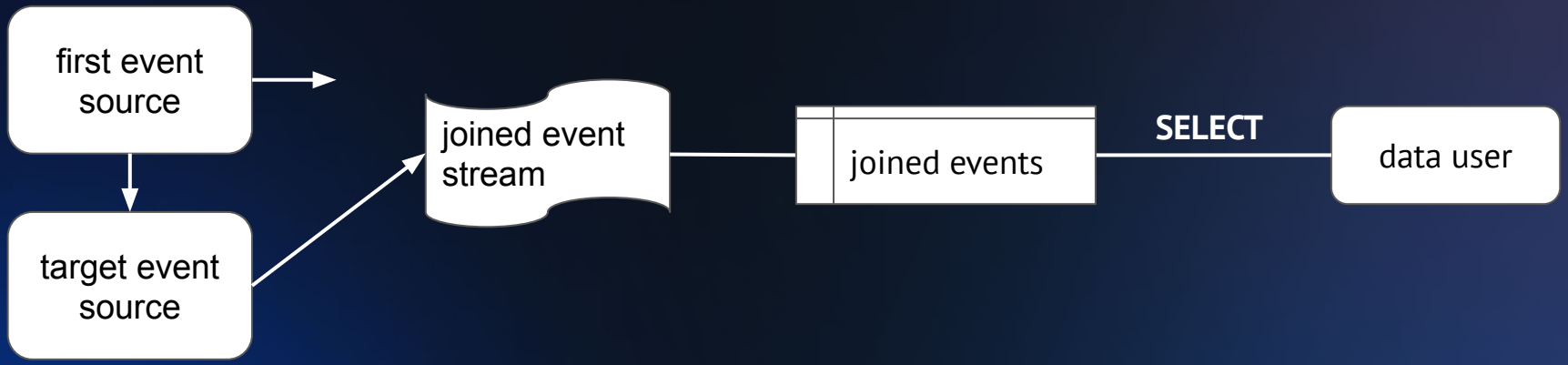
Join. Интерлюдия

product → **stream** → aggregate → runtime



Join. Интерлюдия

product → stream → aggregate → runtime



- Давление требований
- Daily метрики
- Timeliness
- Streaming Ingest
- Big Joins
- Joins. Интерлюдия
- Глубина дат
- Entity State
- High Cardinality
- Наивное решение High Cardinality. Интерлюдия
- Квалификация пользователей
- Пример

5 Глубина дат

- Классическое решение — агрегаты
- Если вы точно знаете, что нужны только недавние даты, то это пространство для оптимизации
- Избегайте закреплённых дат в запросах

6 Entity State

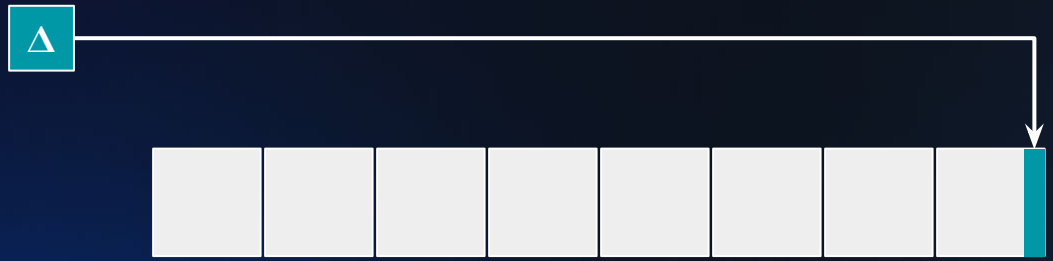
- Придётся делать обе таблицы: log и current state
- Хорошо бы уметь в update
- Change data capture

1B High Cardinality

- Сколько срезов — много?
- Cassandra/ScyllaDB или key-value store

Наивное решение High Cardinality. Интерлюдия

Упорядочено по времени



Наивное решение High Cardinality. Интерлюдия

Упорядочено по high cardinality item

? Квалификация пользователей

- Ограниченный круг пользователей позволяет обеспечивать применение более сложных практик
- BI-системы тоже вносят свой вклад

- Давление требований
- Daily метрики
- Timeliness
- Streaming Ingest
- Big Joins
- Joins. Интерлюдия
- Глубина дат
- Entity State
- High Cardinality
- Наивное решение High Cardinality. Интерлюдия
- Квалификация пользователей
- Пример

Пример. Игры в покер

Источник	*Решение*	*Использование*
Kafka stream	Stream Join	Antifraud
	PostgreSQL	Support
	Партиции по дням	Поиск по пользователю
	GIN-индекс	Глубина до недели

Пример. Игры в покер

Источник	*Решение*	*Использование*
Kafka stream	Stream Join	Antifraud
	PostgreSQL	Support
	Партиции по дням	Поиск по пользователю
	GIN-индекс	Глубина до недели

Пример. Авиабилеты

Источник	*Решение*	*Использование*
Impala DWH	~~Stream Join~~ Reverse ETL	~~Antifraud~~
	PostgreSQL	Support
	Партиции по дням	Поиск по пользователю и другим полям
	~~GIN индекс~~ Много индексов	Глубина до ~~недели~~ года

Пример. Авиабилеты

Источник

Impala DWH

Решение

~~Stream Join~~
Reverse ETL

PostgreSQL

Партиции по дням

~~GIN индекс~~
Много индексов

Использование

~~Antifraud~~

Support

Поиск по пользователю
и другим полям

Глубина до ~~недели~~ года

Пример. Игры в покер

- High cardinality
- Timeliness: ~1 минута
- Ingest: stream
- Join: stream
- Глубина: чаще всего часы, до недели
- Entity State: immutable
- Прямые пользователи: программисты

Пример. Авиабилеты

- High cardinality
- Timeliness: 30 минут
- Ingest: batch
- Join: runtime
- Глубина: до года + запас
- Entity State: есть статус
- Прямые пользователи: поддержка

Вопросы

авиасейлс

Спасибо за внимание!

Артем Дайнеко

Email: dayneko.ab@gmail.com

Telegram: @jamert, личный канал