PostgresPro

Технологии миграции данных из Oracle в СУБД Postgres Pro

Максим Емелин
m.emelin@postgrespro.ru

PostgresPro

О чем будем говорить?

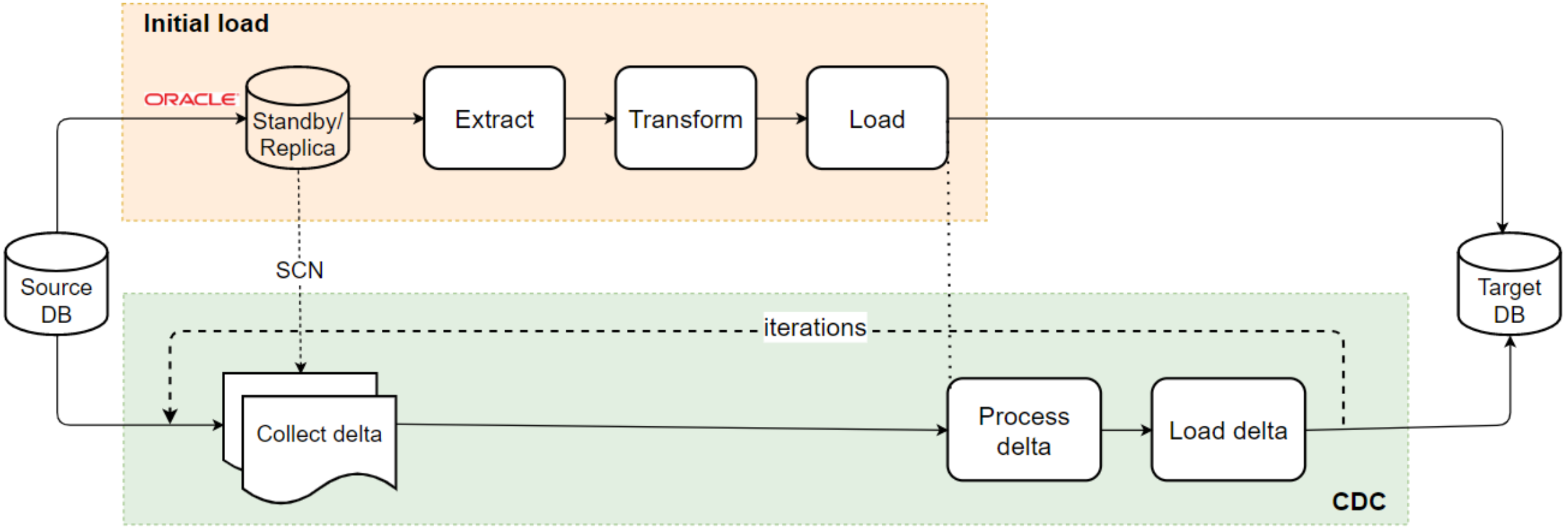
- Общие этапы миграции данных
- Конвертация схемы Oracle
- Копирование данных (initial load)
- Захват данных CDC
- Debezium из коробки
- Как создавать топики
- Особенности и ограничения Debezium for Oracle
- Применение изменений CDC
- Ограничения JDBC Sink Connector
- Верификация качества данных

PostgresPro

Общие этапы миграции данных

1. Конвертация схемы Oracle
2. Копирование данных (initial load)
3. **Захват изменений (change data capture, CDC)**
4. Создание индексов, PK
5. **Применение изменений CDC**
6. Перенос последовательностей, FK
7. Верификация качества данных

Общая схема миграции данных



Конвертация схемы Oracle

Конвертация схемы осуществляется с помощью утилиты **ora2pgpro**

- Конвертация типов данных проходит в соответствии с правилами, описанными в файле lib/Ora2Pg/Oracle.pm
- **ora2pgpro** позволяет конвертировать пакеты Oracle в пакеты Postgres Pro
- Получение скриптов, ручная проверка и исправления
- Перенос последовательностей, создание индексов и констрейнтов

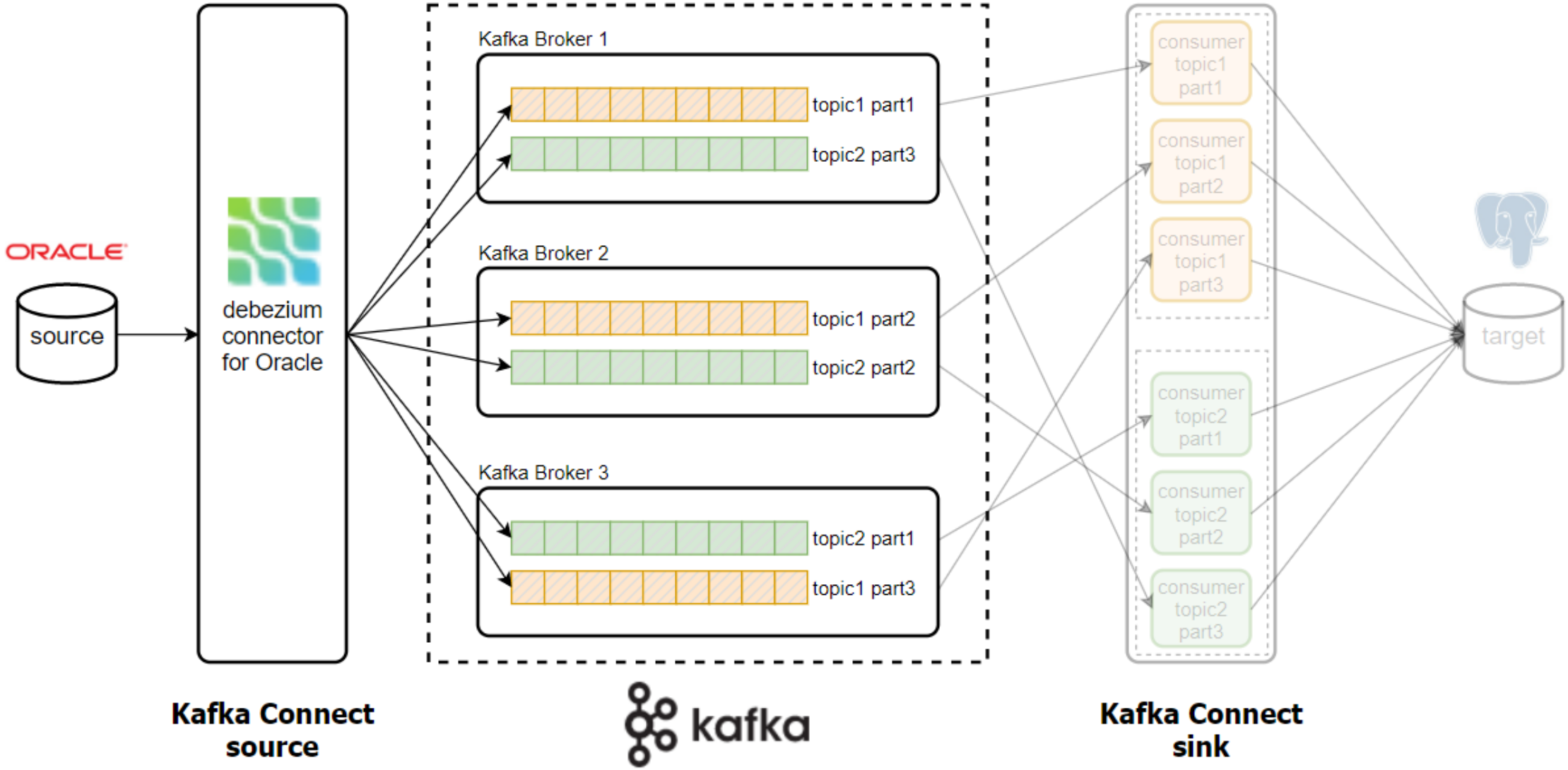
Копирование данных (initial load)

- **Pentaho Data Integration (PDI, aka Kettle)**
  - BLOB не поддерживается
  - нет возобновления копирования данных
- **ora2pgpro**
  - долгое копирование
- **Собственное решение в разработке**
  - ориентировано на предельно быстрое параллельное копирование данных

Захват данных CDC

- **На триггерах**
  - инвазивно
  - не подходит для highload
- **Debezium (LogMiner)**
  - чтение транзакционных журналов
  - оверхед на инфраструктуру
  - не умеет BLOB
- **Собственное решение в разработке**
  - Гибридная функциональность

Захват данных CDC – Debezium



Debezium из коробки

- Коннектор-источник для Kafka Connect
- На текущий момент доступно множество коннекторов к БД: DB2, Microsoft SQL Server, MySQL, **Oracle**, **PostgreSQL,** MongoDB, Cassandra
- Является проектом Red Hat
- Имеет неплохое Community (https://debezium.zulipchat.com)
- CDC, основанный на чтении транзакционных журналов БД
- Умеет делать как initial snapshot, так и отслеживать изменения по таблицам
- Гетерогенен

Как создавать топики?

Если порядок DML не важен (например, в таблицу идут всегда INSERT), можно параллельно вычитывать и делать топик с несколькими партициями

Если порядок DML важен, то все изменения лить в один топик с единственной партицией, поскольку Kafka обеспечивает упорядоченность только в пределах одной партиции

Если нельзя отключить FK на таргет таблицах, то стоит также использовать один топик с единственной партицией

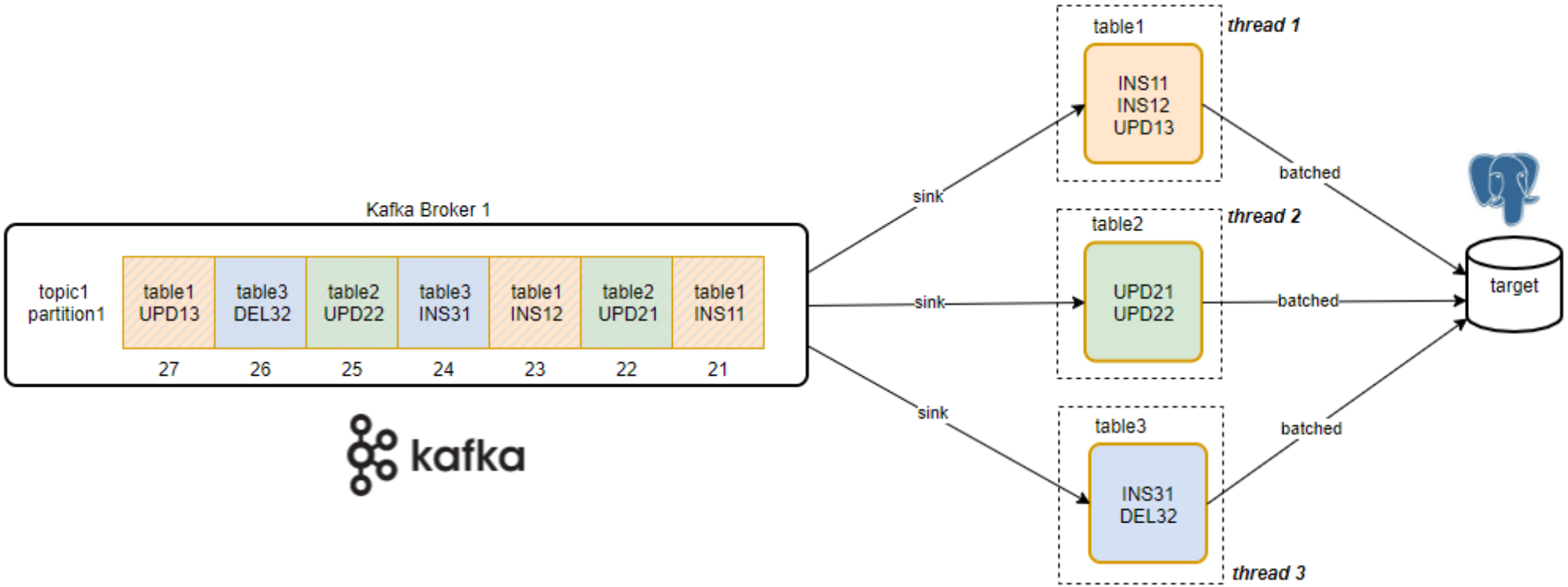
PostgresPro

Особенности и ограничения Debezium для Oracle

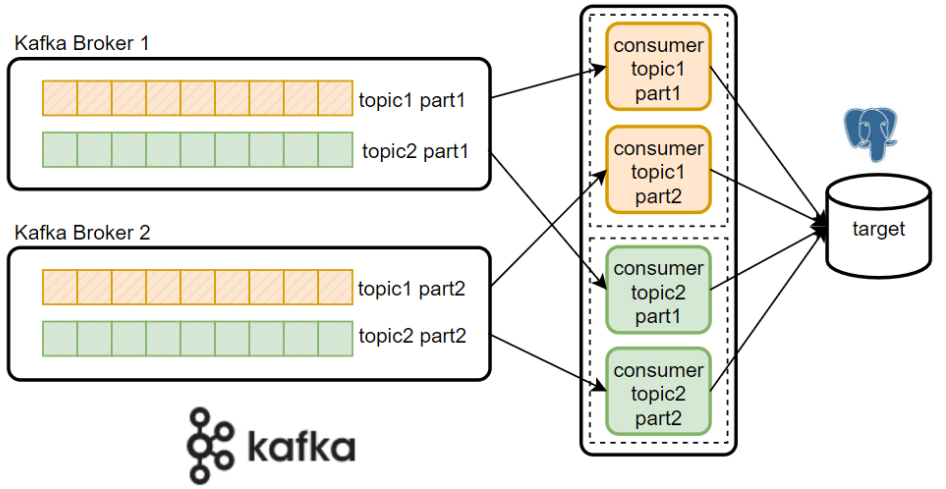
- Поддерживает передачу DDL
- Использует LogMiner или Xstream API
- Таблицы должны иметь PK
- Область памяти для буферизации событий должна быть правильно настроена
- Поддержка LOB в разработке
- Не поддерживаются object types, nested tables, varrays, user-defined types, XML, spatial
- Коннектор не работает на physical/logical standby
- Для стратегии майнинга REDO_LOG_CATALOG должны быть скорректированы параметры Redo
- Supplemental logging должно быть выставлено для таблиц или глобально

Применение изменений CDC

- JDBC Sink connector (в том числе Confluent)
- Написать собственный коннектор (Java)



Ограничения JDBC Sink connector



- Хорошо работает в параллельном режиме, когда запущено несколько консьюмеров по числу партиций топика (или меньше)
- Нельзя использовать, если нужно вычитывать изменения из одного топика с единственной партицией, тем самым, негетерогенен
- Не поддерживает более-менее сложных маппингов и трансформаций

Верификация качества данных

- Механизм Data Quality
- Параллельная работа с несколькими таблицами
- Многопоточность в рамках одной таблицы
- Сравнение по различным критериям
- Использование хэшей, в том числе для CLOB/BLOB
- Собственное решение в разработке

PostgresPro

Спасибо за внимание!

PostgresPro

Q&A