Финальный проект №2 по курсу "Дата-инжиниринг"

Цель проекта

Разработать распределённый ETL-пайплайн для аналитики e-commerce платформы с поддержкой мультимодальных данных. Проект должен включать генерацию событий, оркестрацию в Airflow, хранение данных в PostgreSQL, ClickHouse и Neo4j, построение аналитических витрин в ClickHouse на основе данных из нескольких источников и визуализацию в Tableau Public. Архитектура должна демонстрировать современный подход к построению Data Warehouse с разделением ответственности между различными типами хранилищ.

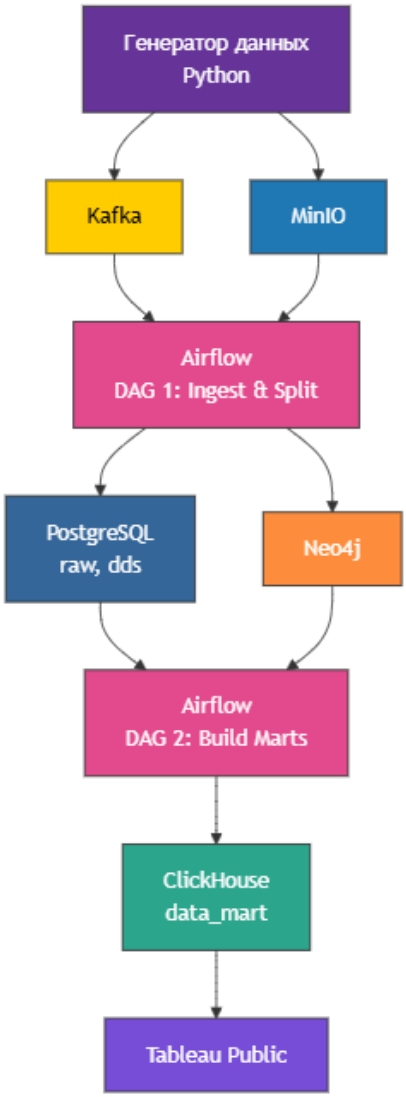

Рис. 1. Схема ETL

Архитектура (обязательные компоненты)

Проект должен включать следующие компоненты:

- Генератор данных (Python)
- Источник: Kafka и MinIO
- Airflow (оркестрация)
- PostgreSQL (DWH, нормализация)
- Neo4j (граф связей)

- ClickHouse (аналитические витрины)
- Tableau Public (визуализация)

Выбирает один из двух вариантов и реализует пайплайн от начала до конца.

Предметная область

Вы разрабатываете аналитическую систему для социальной сети, где пользователи могут создавать контент, взаимодействовать друг с другом, участвовать в сообществах, делиться медиа и влиять на аудиторию. Система должна учитывать не только активность, но и социальные связи, вирусное распространение контента и поведение сообществ. Основные сущности и потоки данных:

Группа	**Сущности**
Пользователи	users, user_profiles, user_settings, user_privacy, user_status (online/offline), user_badges
Социальные связи	friends, followers, subscriptions, blocks, mutes, close_friends
Контент	posts, stories, reels, comments, replies, shares, likes, reactions (like, love, wow и т.д.)
Медиа	photos, videos, albums, attachments, thumbnails
Сообщества и группы	communities, groups, group_members, group_moderators, community_topics, pinned_posts
Монетизация	ads, ad_impressions, ad_clicks, sponsorships, donations, subscriptions_revenue

Итоговые метрики, которые могут быть рассчитаны в Clickhouse:

Общая активность в сети. Ключевые метрики платформы:

- Общее количество постов — суммарное число созданных публикаций. Показывает уровень контент-активности.
- Количество комментариев и реакций — общее число взаимодействий. Характеризует вовлечённость.
- Среднее количество взаимодействий на пост — отношение всех реакций и комментариев к числу постов. Отражает качество контента.
- Количество активных пользователей за день (DAU) — число уникальных пользователей, совершивших хотя бы одно действие.
- Количество активных пользователей за 7 дней (WAU) — аналогично, за неделю.
- Соотношение DAU/WAU — показатель удержания и вовлечённости.

Социальные графы и влияние. Анализ связей и влияния:

- Глубина дружбы — количество уровней в цепочке дружбы (например, друг → друг друга → друг друга друга). Позволяет оценить связность сети.
- Количество друзей и подписчиков — среднее число связей на пользователя.
- Центральность пользователя — метрика из Neo4j (например, PageRank), показывающая, насколько пользователь "важен" в сети.
- Количество репостов и цитат — сколько раз контент пользователя был подхвачен другими. Показывает вирусность.
- Скорость распространения контента — как быстро пост достигает 1000 просмотров. Характеризует "вирусный" потенциал.

Сообщества и темы. Аналитика групп и обсуждений:

- Топ-сообщества по росту — группы с наибольшим приростом участников.
- Активность в сообществах — количество постов, комментариев, реакций по группам.
- Количество модераторов и жалоб — показатель нагрузки на модерацию.
- Самые обсуждаемые темы — топики с наибольшим числом комментариев.
- Коэффициент вовлечённости в группе — отношение активных участников к общему числу.

Технические требования

1. Инфраструктура
    1) Необходимо поднять все технологии для работы в docker-контейнерах
    2) репозитории должен лежать образ, по которому можно запустить инфраструктуру
2. Генерация данных
    1) События генерируются **каждую минуту**
    2) Формат: **JSON**
    3) Реалистичные значения
3. Выбор источника
    1) Вариант A: *Kafka* — события в топик
    2) Вариант B: *MinIO* — JSON-файлы с именем по времени
4. *Airflow*
    1) Этапы: чтение → валидация → очистка → загрузка → витрины
5. Валидация
    1) *Pydantic* или ручная проверка
    2) Типы, диапазоны
    3) Удаление дубликатов
6. Обработка ошибок
    1) *try-except*
    2) Логирование (*INFO, ERROR*)
7. Алертинг
    1) Уведомления при успехе и ошибке
    2) Telegram или Email
    3) *on_success_callback / on_failure_callback*
8. Структура кода
    1) Декомпозиция по папкам: *generator, dags, utils, sql, config* или другим
    2) Комментарии к коду
9. DWH
    1) Должна использоваться хоть одна из моделей в *PostgreSQL* для хранения данных – Звезда, Снежинка или Data Vault
    2) В *Clickhouse* должны быть витрины в денормализованом формате для предоставления витрин Аналитикам и дашбордам
10. Визуализация
    1) Tableau Public
    2) Дашборд
11. Документация
    1) README.md
    2) Описание архитектуры и описание всех моделей (*draw.io*)
    3) Скриншоты

Требования к сдаче

- Публичный репозиторий
- Видео (3–5 мин): работа пайплайна, дашборд