Полная задача проекта

Вы работаете в e-commerce-проекте. Бизнес-юнит получил доступ к логам посещения портала для большого пула пользователей. Вашей команде нужно подготовить данные для первичного анализа и собрать несколько графиков в дашборд. Финальный результат вашей работы, который вы покажете наставникам, а они оценят — дашборд с графиками и сводкой по данным.

Вам нужно:

- спроектировать хранилище в PostgreSQL со слоями, которые вы посчитаете нужными;
- написать алгоритмы для создания витрин(ы) и реализовать с помощью Spark-джоб;
  - возможно, добавить это всё в оркестратор;
- сделать на основе витрин(ы) дашборд в Metabase.

Что есть на входе

- доступы в Spark, Postgres (на каждую команду свои юзеры) и в Airflow;
- данные в виде файла в S3 (около 250Мб на один день).

А что за данные?

Кликстрим. Несколько тысяч одинаковых по структуре JSON-записей, которые получились из сбора данных о пользовательском поведении на портале: какие действия делали → какие события фиксировали системы аналитики.

▼ Пример события


```
{
  "event_timestamp": "2020-07-05 14:32:45.407110",
  "event_type": "pageview",
  "page_url": "http://merch.practicum.ru/home",
  "page_url_path": "/home",
  "referer_url": "www.instagram.com",
  "referer_url_scheme": "http",
  "referer_url_port": "80",
  "referer_medium": "internal",
  "utm_medium": "organic",
  "utm_source": "instagram",
  "utm_content": "ad_2",
  "utm_campaign": "campaign_2",
  "click_id": "b6b1a8ad-88ca-4fc7-b269-6c9efbbdad55",
  "geo_latitude": "-25.54073",
  "geo_longitude": "152.70493",
  "geo_country": "AU",
  "geo_timezone": "Australia/Brisbsane",
  "geo_region_name": "Maryborough",
  "ip_address": "209.139.207.244",
  "browser_name": "Firefox",
  "browser_user_agent": "Mozilla/5.0 (Macintosh; U; PPC Mac OS X 10_5_5; rv:1.9.6.20) Gecko/2012-06-06 09:24:19 Firefox/3.6.20",
  "browser_language": "tn_ZA",
  "os": "Android 2.0.1",
  "os_name": "Android",
  "os_timezone": "Australia/Brisbane",
  "device_type": "Mobile",
  "device_is_mobile": true,
  "user_custom_id": "vsnyder@hotmail.com",
  "user_domain_id": "3d648067-9088-4d7e-ad32-45d009e8246a"
}
```

У вас будет по несколько тысяч таких сообщений «в час» относительно поля `event_timestamp` в самих событиях. Все действия совершены в рамках одной страны.

Данные будут доступны по ссылке, в сыром виде они займут несколько сотен мегабайт.

Что нужно сделать

В задаче есть базовая и усложнённая часть. Базовую нужно сделать в любом случае, а вот за дополнительную можете браться, если после базовой будет время, силы и желание. Различаются они в том, какие графики, а соответственно, витрины и процессы нужно создать.

Процесс решения для обеих задач общий:

1. Спроектировать систему так, чтобы можно было при необходимости всё пересчитать.
2. Создать хранилище данных в Postgres: количество, слои и модели данных — на ваше усмотрение.
3. Построить пайплайн, который выгрузит данные из источников и при необходимости обработает и загрузит в витрину.
4. Построить графики и дашборд в Metabase.
5. Все принятые решения по данным зафиксировать в общий документ команды. Туда же заложить описание архитектуры решения — так, будто вы отдаёте его реальным бизнес-пользователям.

Базовая часть

Бизнес уже сформулировал визуализации, которые точно должны быть в дашборде:

1. распределение событий по часам;
2. количество купленных товаров в разрезе часа;
3. топ-10 посещённых страниц, с которых был переход в покупку — список ссылок с количеством покупок.

Вам нужно самостоятельно определиться с остальными вводными, например, что считать покупкой, как разбивать на часы или как формируется ссылка. Главное пожелание — зафиксируйте в документации решения, которые примете об обработке данных. И не бойтесь их менять при необходимости.

Усложнённая часть

Ниже в несортированном порядке перечислены идеи, которые предложил бизнес-юнит. Вам не нужно делать всё. Если у вас есть интересная идея визуализации, которая поможет бизнесу принять решение, воплотите её.

Идеи:

- Проанализировать и визуализировать покупки по источникам. В данных заложены источники и рекламные кампании, из которых пользователи переходили на портал. Создайте визуализацию, показывающую процентное соотношение пользователей.
- Добавить к графикам из базовой части возможность смотреть их в разрезе браузеров (Chrome / Firefox / InternetExplorer / Safari) и платформ (телефон/другие устройства).
- Графики, которые по вашему усмотрению смогут лучше всего показать, какие сегменты пользователей больше всего покупают. Определить параметры сегментации предстоит вам. Они могут совпадать с другими предложенными вариантами в этом списке.

На входе — те же данные, но, возможно, есть дополнительные источники. Уточните в канале проекта.

> ❗ Постарайтесь использовать существующие структуры и добавлять новые визуализации не ломая то, что сделали для базового решения.

Как делать или требования к решению

- **Самостоятельность в проектировании.** В задаче специально нет подробных вводных о том, как располагать данные в хранилище, как называть таблицы и как писать алгоритмы обработки и что они должны делать. Вам нужно определиться с этим самостоятельно.
- **Гибкость решения.** В него нужно заложить возможность пересчитать все витрины и, соответственно, визуализации, так как данные в процессе работы могут добавляться или меняться. При этом их структура всегда будет одинаковой.
- **Нужна ли Spark-джоба?** Легко возразить, что такую задачу можно решить без инструментов для обработки больших данных — если данные предоставлены в конечном виде, их можно загрузить в скрипт и обработать.

  И это так, если бы вам нужно было подготовить единоразовый отчёт — в случае с обычной одноразовой локальной обработкой действительно достаточно один раз скачать данные на компьютер и сделать отчёт, произведя все вычисления локально. Вам же нужно создать дашборд, которым будут пользоваться аналитики и другие люди. Для полноценного дашборда полезно построить хранилище данных и организовать обработку и подготовку витрин для визуализации в нём.

  К тому же, если проект окажется перспективным, подобные витрины можно будет использовать для алгоритмической персонализации контента и аналогичных задач.