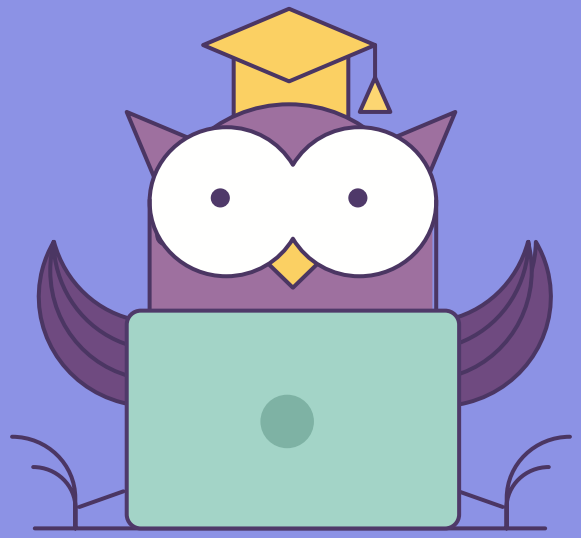
ОНЛАЙН-ОБРАЗОВАНИЕ

Меня хорошо слышно && видно?



Напишите в чат, если есть проблемы!

Ставьте + если все хорошо

OTUS

GCP, AWS, Azure

Облачные платформы

Матешук Егор

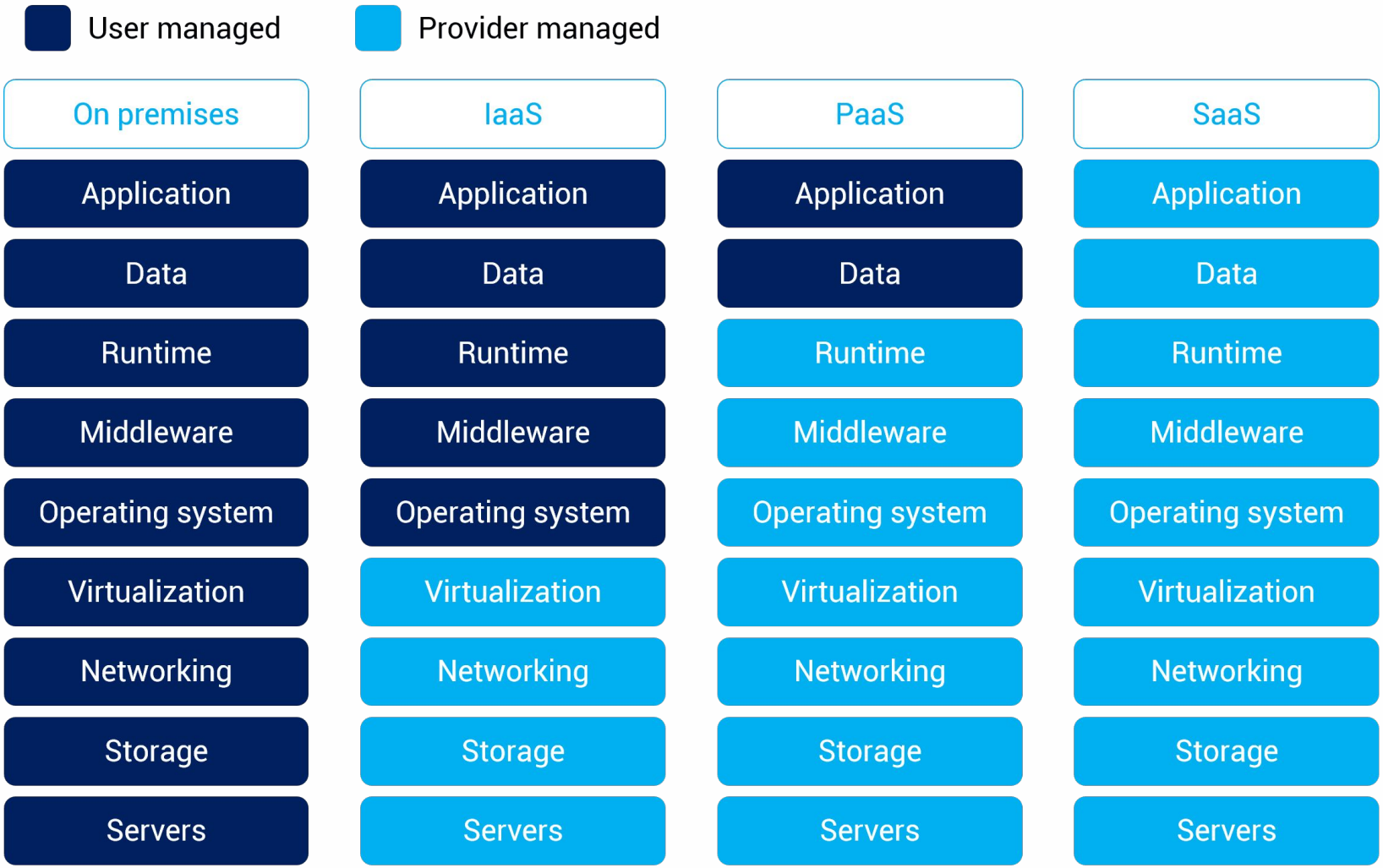
Цели сегодняшнего занятия

- Обсудить, что такое облачные платформы
- Узнать о сервисах для дата-инженеров
- Выяснить, в каких случаях нужно использовать облака
- Построить первый процесс обработки данных в GCP

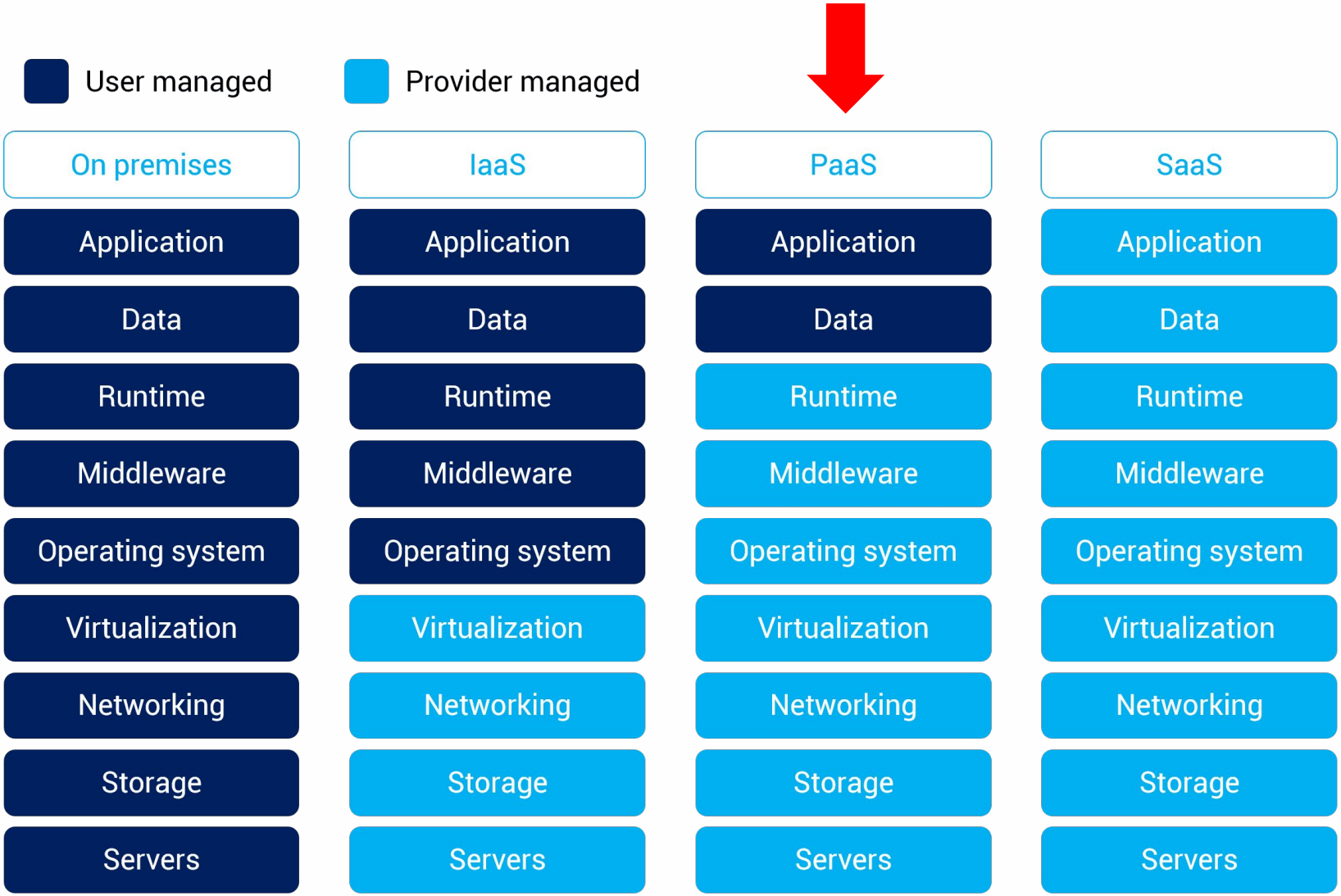
01

Существующие облачные платформы

Что такое облачные платформы



Что такое облачные платформы



Главные облачные платформы

- Google Cloud Platform
- Amazon Web Services
- Microsoft Azure

02

Облачные инструменты для дата-инженеров

Главные облачные платформы

- Object storage
- Message queue
- Data ingestion tool
- Compute engine
- MPP SQL database
- Operational database
- Orchestration tool

Object storage

- Cloud Storage (GCP)
- S3 (AWS)

Message queue

- Pub/Sub (GCP)
- Kinesis / Managed Kafka (AWS)

Data ingestion tool

- Cloud Data Fusion (GCP)
- Glue (AWS)

Compute engine

- Dataflow / Dataproc (GCP)
- EMR (AWS)

MPP SQL database

- BigQuery (GCP)
- Redshift / Athena (AWS)

Operational database

- BigTable (KV), Spanner (SQL) (GCP)
- DynamoDB (KV), Aurora (SQL) (AWS)
- … и множество других под конкретные паттерны хранения и доступа

Orchestration tool

- Composer (GCP)
- Data Pipeline (AWS)

03

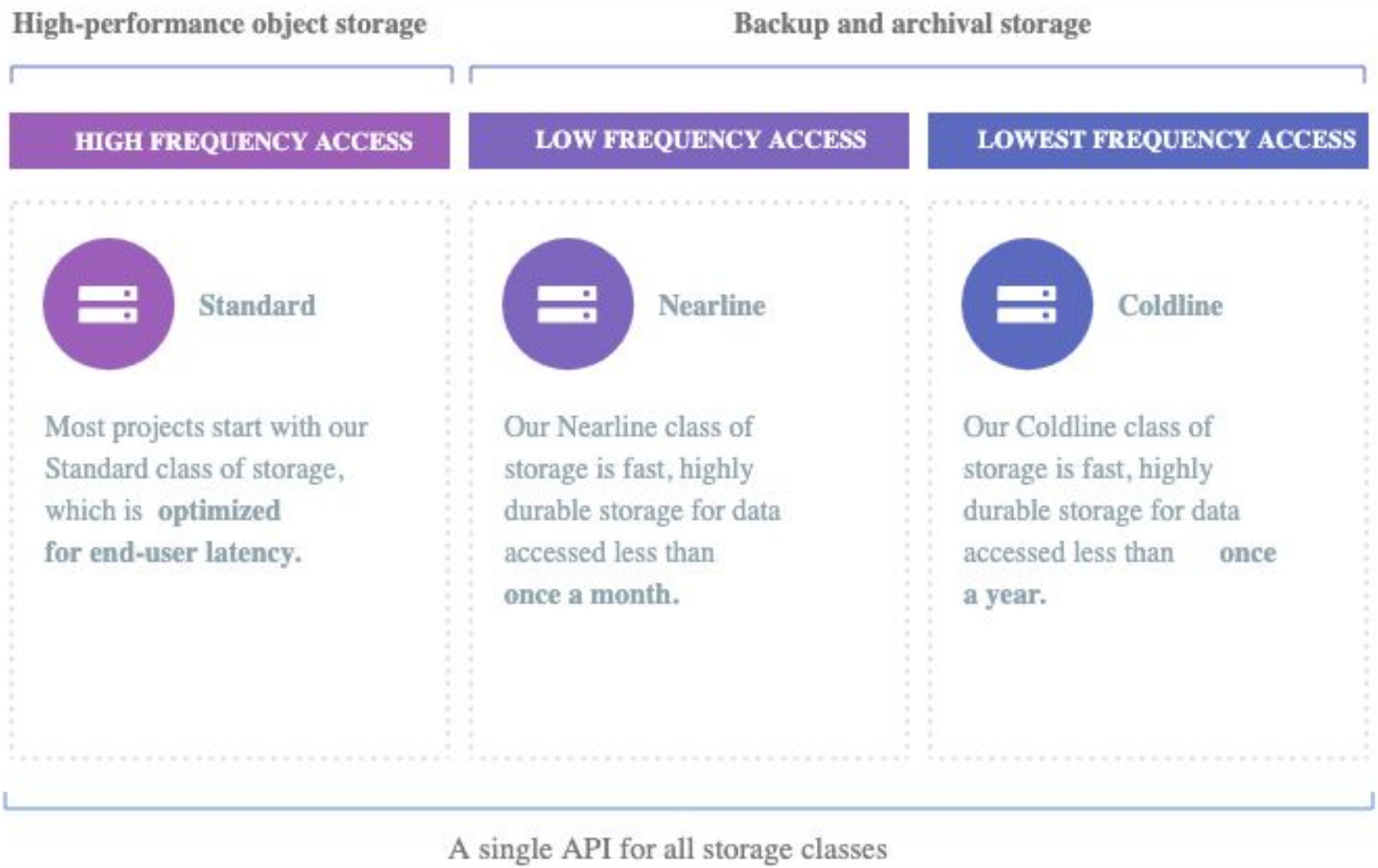
Подробнее о сервисах Google Cloud Platform

- Облачное гео-распределенное хранилище для файлов (не путать с Google Drive)
- Основной элемент - bucket (сегмент в русской версии)

Большая часть настроек задается на уровне этих самых бакетов

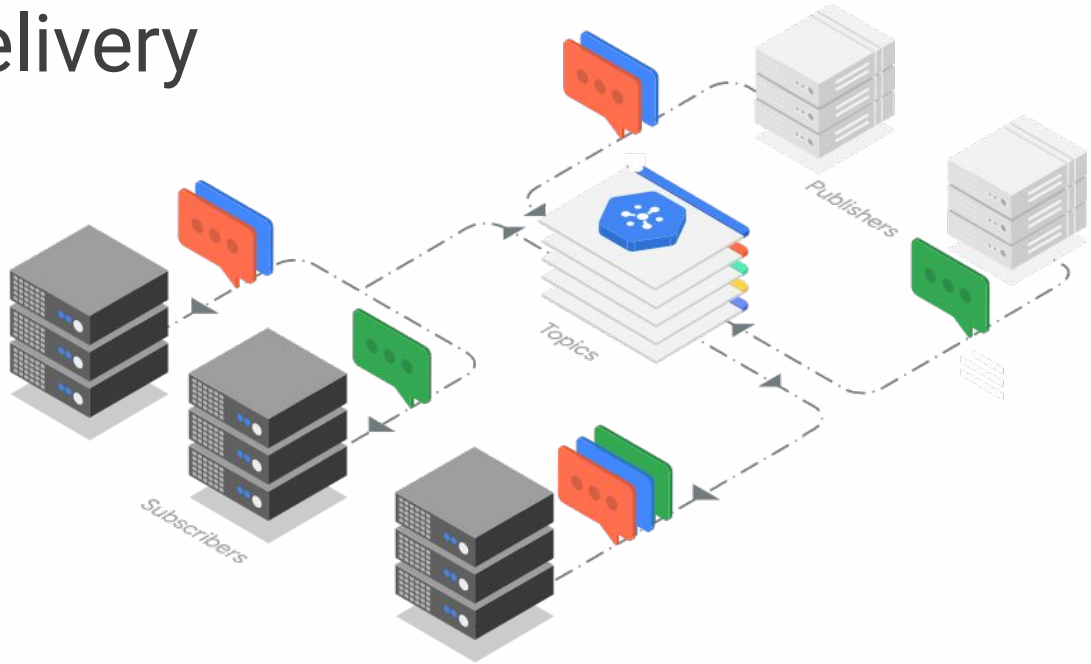
- Регион
- Версионирование
- Класс хранилища
- Политики хранения
- Шифрование

Cloud Storage - Storage classes

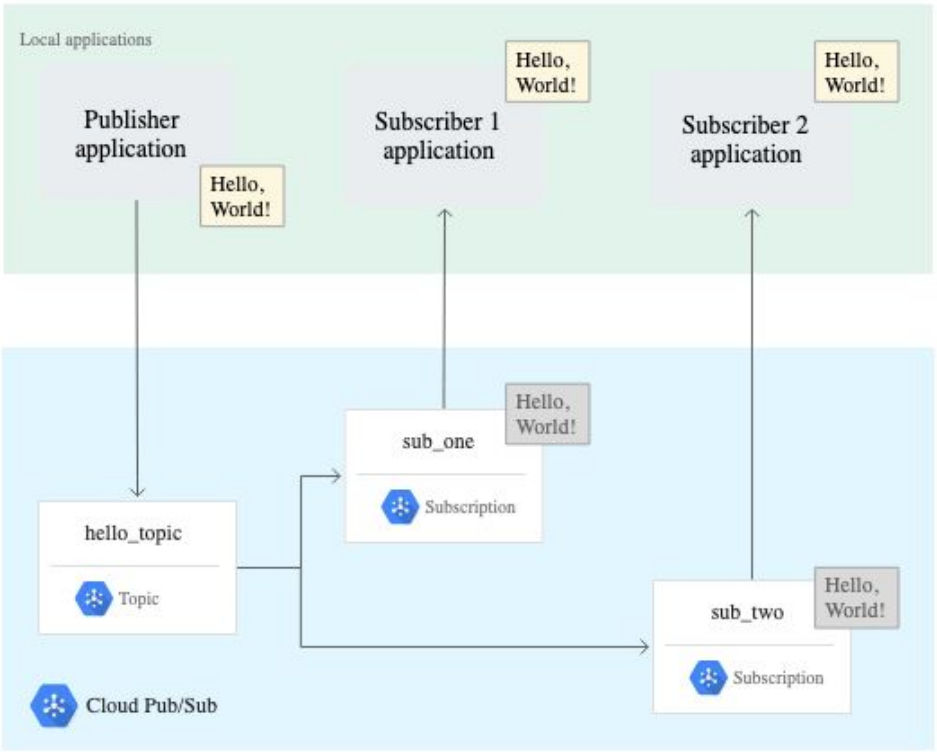


Pub/Sub

- Очередь сообщений
- Возможность репроцессинга
- Глобальная репликация
- Exactly-once delivery
- Auto-scaling

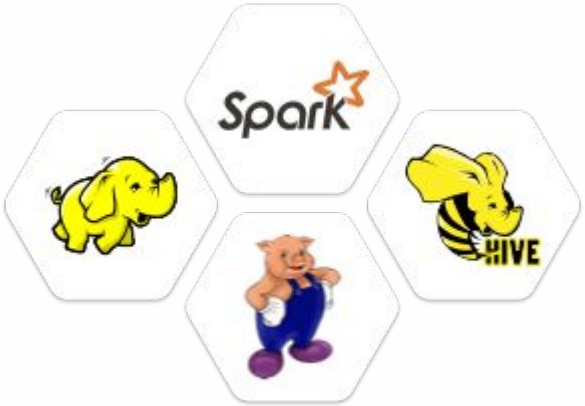


- Основные элементы - topic, subscription
- Нет партиций (как в Kafka)



Dataproc

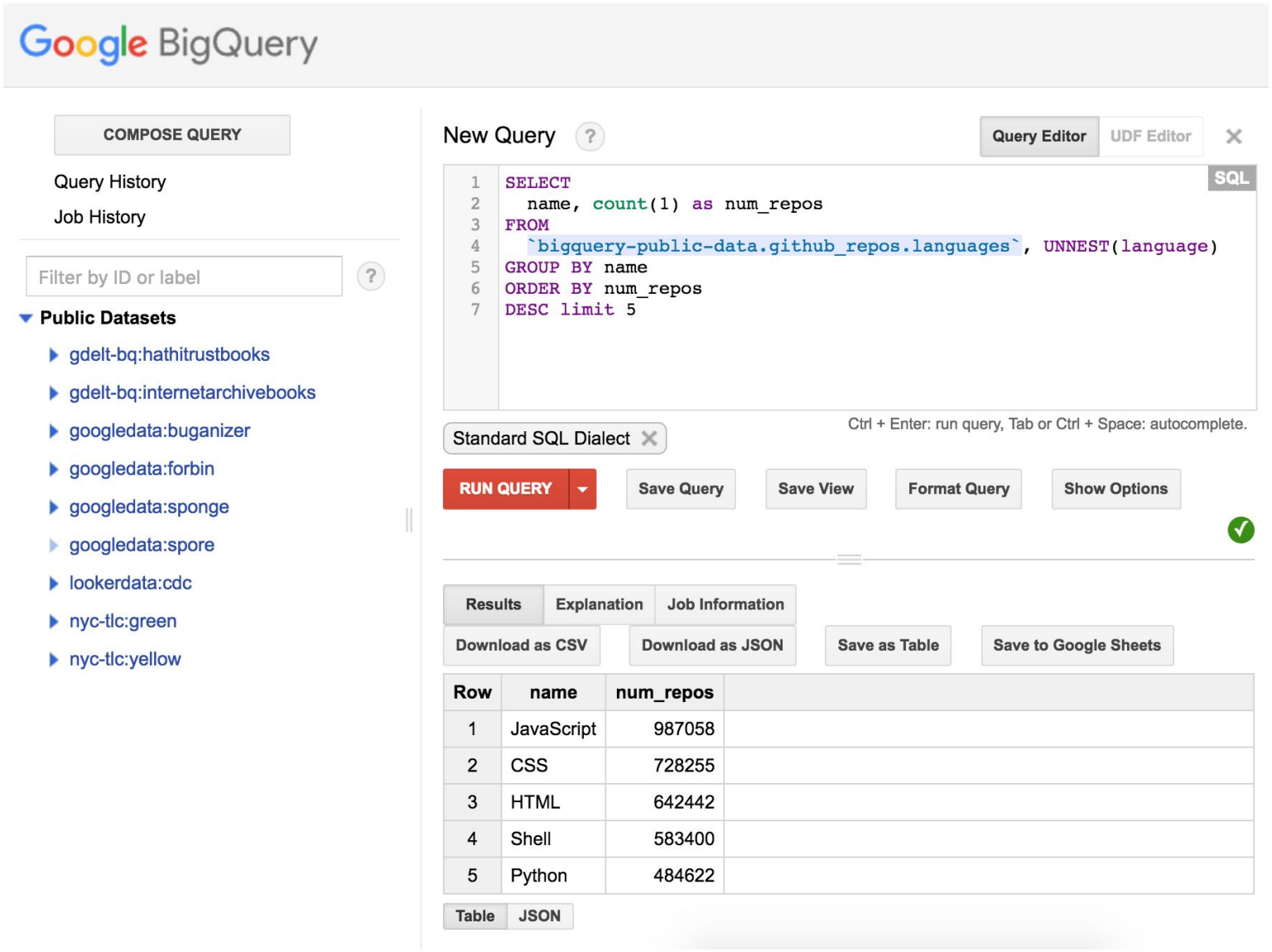
- Managed Spark + Hadoop + Hive
- Кластер по запросу
- Интеграция с экосистемой GCP



- Также известный как Apache Beam
- Использует концепцию универсального подхода к batch и streaming процессингу данных
- Запускается на нескольких бэкендах, включая основной - сервис Google Dataflow

- SaaS-решение из экосистемы Google Cloud
- Оплата за объем обработанных данных
- Поддерживает слабо-структурированные данные
- Имеет встроенную среду для работы с запросами

BigQuery


Google BigQuery
COMPOSE QUERY
Query History
Job History
Filter by ID or label
Public Datasets
gdelt-bq:hathitrustbooks
gdelt-bq:internetarchivebooks
googledata:buganizer
googledata:forbin
googledata:sponge
googledata:spore
lookerdata:cdc
nyc-tlc:green
nyc-tlc:yellow
New Query
Query Editor
UDF Editor
SQL
1 SELECT
2 name, count(1) as num_repos
3 FROM
4 `bigquery-public-data.github_repos.languages`, UNNEST(language)
5 GROUP BY name
6 ORDER BY num_repos
7 DESC limit 5
Ctrl + Enter: run query, Tab or Ctrl + Space: autocomplete.
Standard SQL Dialect
RUN QUERY
Save Query
Save View
Format Query
Show Options
Results
Explanation
Job Information
Download as CSV
Download as JSON
Save as Table
Save to Google Sheets
Row name num_repos
1 JavaScript 987058
2 CSS 728255
3 HTML 642442
4 Shell 583400
5 Python 484622
Table
JSON

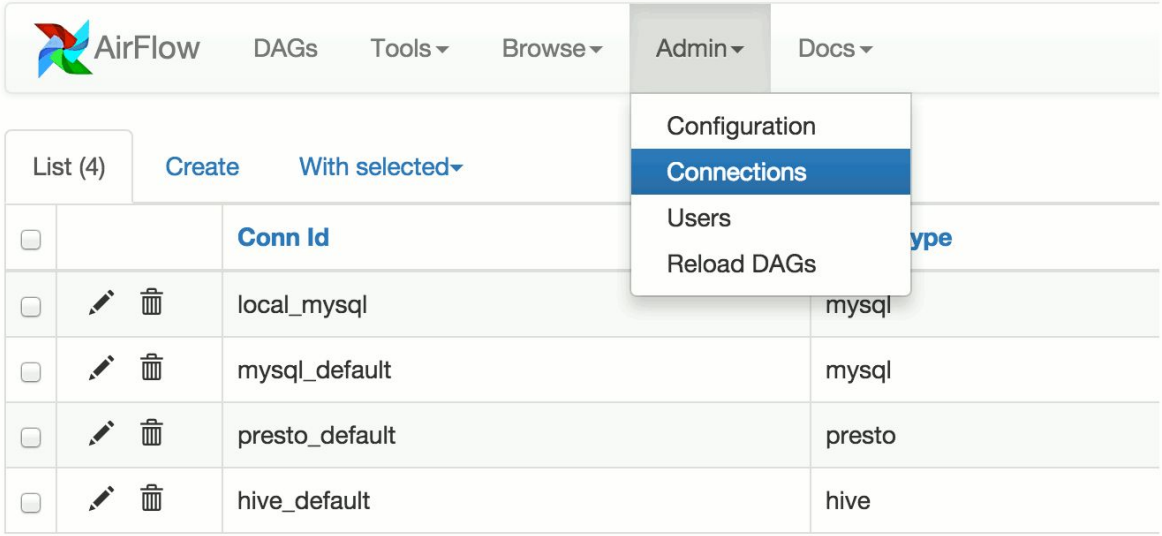
- Google Composer

  = Managed Apache Airflow
- Развертывается в виде кластера из нескольких машин с хранилищем в GCS

Composer (Airflow)

Apache Airflow — платформа для автоматического управления задачами, их расписанием и мониторингом.

Изначально был разработан в AirBNB.



Composer (Airflow)

- Задачи описываются на Python
- Поддерживает сложные процессы из нескольких задач с зависимостями, ветвлением, условиями
- Удобный интерфейс
- Может масштабироваться с использованием Celery
- Легко расширяется под конкретные задачи

04

Когда нужны облака

Когда подходят облака

- Отсутствие DevOps-компетенций в команде
- Работа с зарубежными данными
- Гео-распределенные сервисы
- Автономные команды

Дополнительные плюсы облака

- Более точная оценка затрат
- Ускорение циклов разработки
- “Защита из коробки”
- Интеграция продуктов

Интересный факт

В 2015 году Spotify решили полностью отказаться от собственной инфраструктуры и перенести сервисы и процессы обработки данных на GCP

Мотивация для перехода с собственной инфраструктуры на облако

- Избавление от нерелевантной деятельности
- Экономия на поддержке
- Миграция с Kafka на Pub/Sub
- Миграция со стека Spark + Hadoop на Dataflow
- Использование BigQuery

Spotify case

Для понимания масштаба

- Самый большой Hadoop-кластер в Европе (~2500 узлов)
- 20 000 ежедневных задач на кластере
- На скорости 160 Gbit/s копирование данных без учета новых изменений заняло бы два месяца

В итоге

- Миграция шла около 12 месяцев
- Многие пайплайны переносили по пути

  Spark > Dataproc > Dataflow
- В процессе написали Scala-фреймворк для работы с Dataflow (Scio)

05

Практика

Заполните, пожалуйста, опрос в личном кабинете!

Полезные материалы

- Рассказ о миграции Spotify
- The Road to Scio
- Google Cloud SDK
- Google Cloud Storage CLI
- Сопоставление сервисов GCP и Azure

Егор Матешук

egor@mateshuk.com

Спасибо
за внимание!