Workshop №3

Kafka

Яндекс Практикум @moneretin Михаил Неретин

План воркшопа

1. Kafka, что это за фрукт и с чем его едят?

План воркшопа

1. Kafka, что это за фрукт и с чем его едят?
2. Принцип работы.

План воркшопа

1. Kafka, что это за фрукт и с чем его едят?
2. Принцип работы.
3. Поработаем с Kafka с помощью Python.

План воркшопа

1. Kafka, что это за фрукт и с чем его едят?
2. Принцип работы.
3. Поработаем с Kafka с помощью Python.
4. Изучим формат данных Avro и причем тут Schema Registry.

План воркшопа

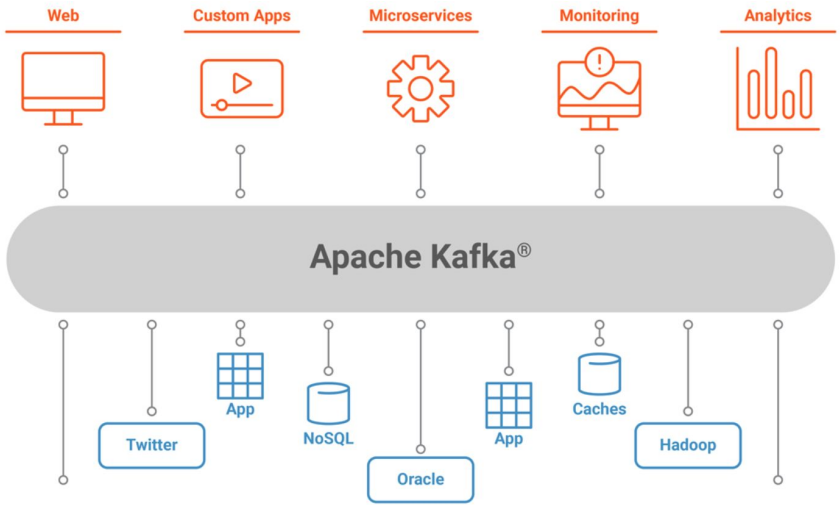
1. Kafka, что это за фрукт и с чем его едят?
2. Принцип работы.
3. Поработаем с Kafka с помощью Python.
4. Изучим формат данных Avro и причем тут Schema Registry.
5. Поработаем с Avro и Kafka с помощью Python.

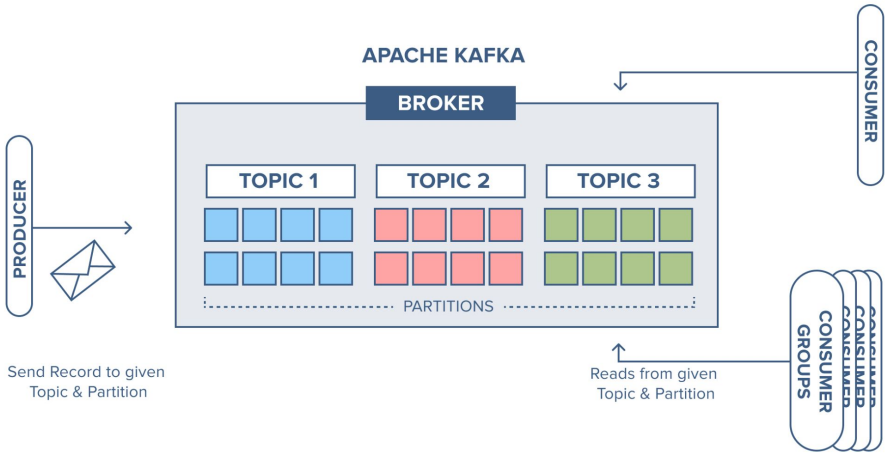
Что такое Kafka?

Kafka — это распределенная потоковая платформа

Что такое Kafka?

Kafka — это распределенная потоковая платформа





Что такое Kafka?

Что такое Kafka?

Kafka — это распределенный ленточный конвейер

Как устроена Kafka?

Вид обработки сообщений — publish/subscribe (Pub/Sub).

Как устроена Kafka?

Вид обработки сообщений — publish/subscribe (Pub/Sub).

Pub/Sub — паттерн проектирования передачи сообщений.

Как устроена Kafka?

Вид обработки сообщений — publish/subscribe (Pub/Sub).

Pub/Sub — паттерн проектирования передачи сообщений.

Ключевые особенности:

Как устроена Kafka?

Вид обработки сообщений — publish/subscribe (Pub/Sub).

Pub/Sub — паттерн проектирования передачи сообщений.

Ключевые особенности:

- **Publisher** (издатель) не отправляет сообщение конкретному потребителю, а каким-то образом классифицирует их.

Как устроена Kafka?

Вид обработки сообщений — publish/subscribe (Pub/Sub).

Pub/Sub — паттерн проектирования передачи сообщений.

Ключевые особенности:

- **Publisher** (издатель) не отправляет сообщение конкретному потребителю, а каким-то образом классифицирует их.
- **Subscriber** (подписчик) подписывается на определенные классы сообщений.

Как устроена Kafka?

Вид обработки сообщений — publish/subscribe (Pub/Sub).

Pub/Sub — паттерн проектирования передачи сообщений.

Ключевые особенности:

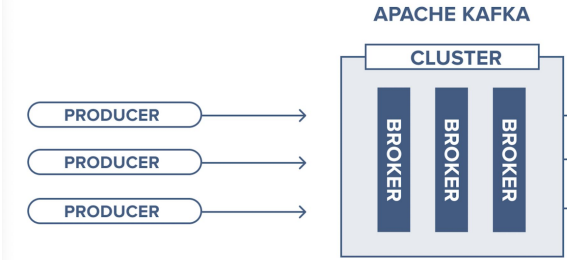
- **Publisher** (издатель) не отправляет сообщение конкретному потребителю, а каким-то образом классифицирует их.
- **Subscriber** (подписчик) подписывается на определенные классы сообщений.
- **Broker** (брокер) — центральный пункт публикации сообщений.

Как устроена Kafka?

Producer — пишет данные в Kafka (тот самый publisher)

Как устроена Kafka?

Producer — пишет данные в Kafka (тот самый publisher)

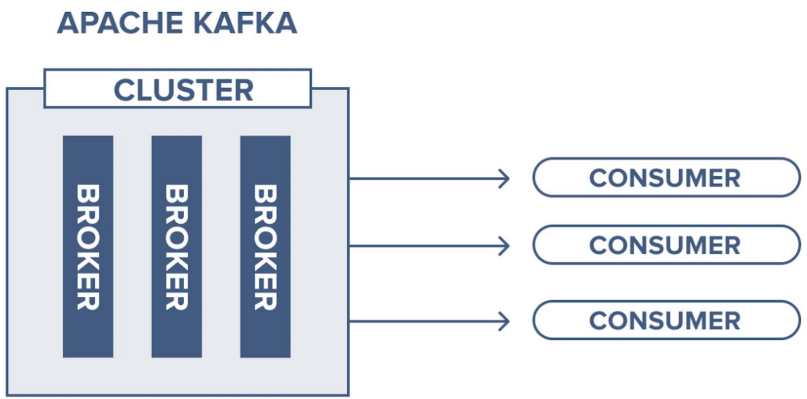


Как устроена Kafka?

Consumer — читает данные из Kafka (тот самый subscriber)

Как устроена Kafka?

Consumer — читает данные из Kafka (тот самый subscriber)



Как устроена Kafka?

- **Сообщение** (message) — базовая единица данных в Kafka.

Как устроена Kafka?

- **Сообщение** (message) — базовая единица данных в Kafka.
- Сообщение состоит из key и value, где key выступает элементом метаданных.

Как устроена Kafka?

- **Сообщение** (message) — базовая единица данных в Kafka.
- Сообщение состоит из key и value, где key выступает элементом метаданных.
- Сообщения организованы в **топики** (topics).

Как устроена Kafka?

- **Сообщение** (message) — базовая единица данных в Kafka.
- Сообщение состоит из key и value, где key выступает элементом метаданных.
- Сообщения организованы в **топики** (topics).
- Топики разбиты на **партиции** (partitions), где и хранят свои сообщения.

Как устроена Kafka?

Сообщение (message) — базовая единица данных в Kafka.

Сообщение состоит из key и value, где key выступает элементом метаданных. Бутылка с соком - value; Сок апельсиновый - key.

Как устроена Kafka?

Сообщения организованы в **топики** (topics).

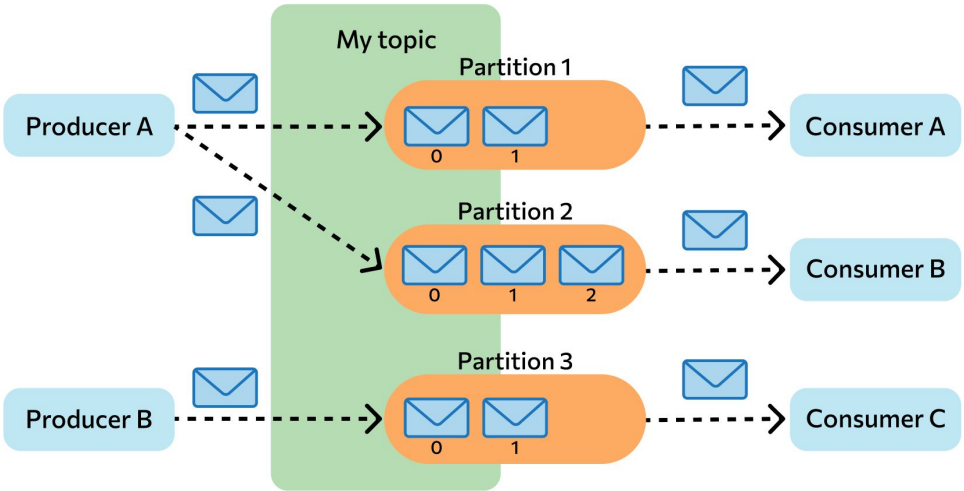
Топик - бананы. Все бананы на конвейерах - это один банановый топик.

Как устроена Kafka?

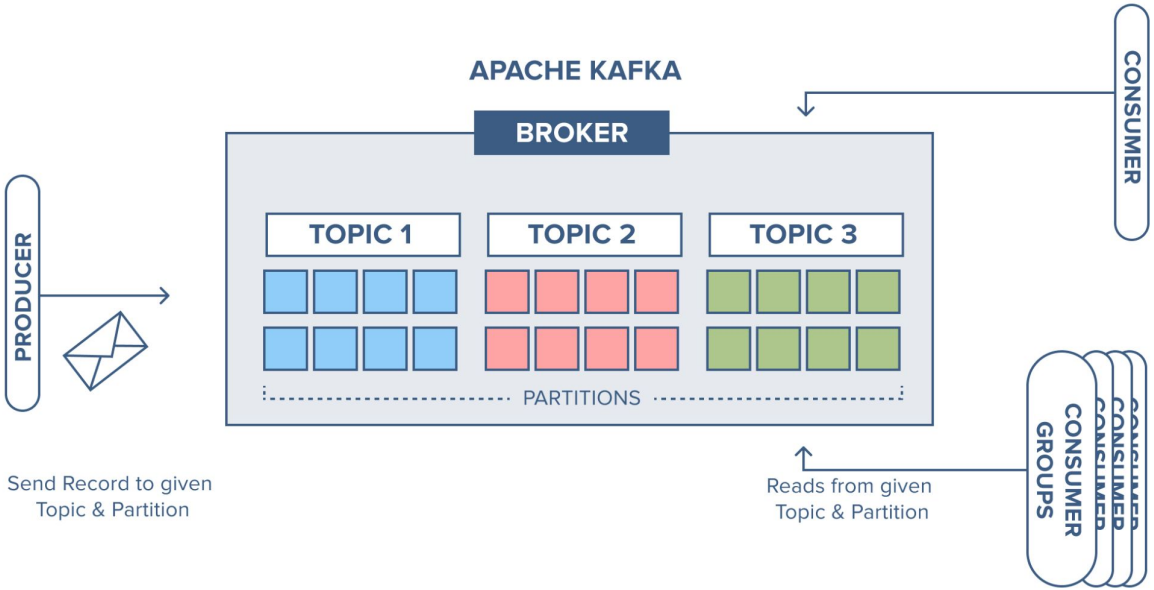
Топики разбиты на **партиции** (partitions), где хранят свои сообщения.

Партиция - это есть ленточный конвейер.

Как устроена Kafka?



Как устроена Kafka?

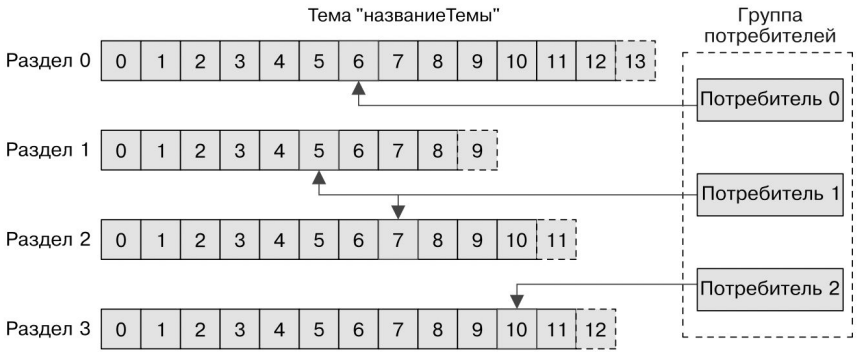


Как работает producer?

- **Producer** — просто пишет данные в топик.

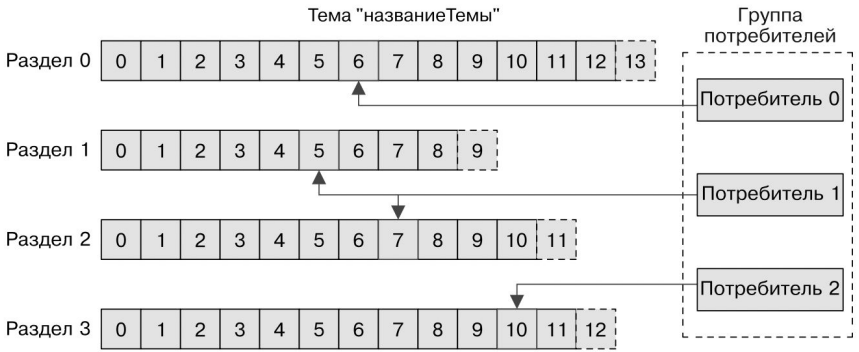
Как устроена Kafka?

- **Consumer** — читает сообщения и отслеживает какие сообщения прочитал, запоминая их *offset* (смещение).



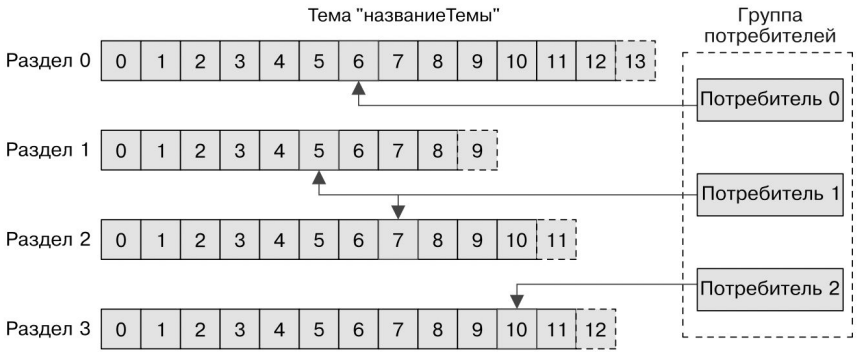
Как устроена Kafka?

- **Consumer** — читает сообщения и отслеживает какие сообщения прочитал, запоминая их *offset* (смещение).
- Consumers могут объединяться в *consumers group*.



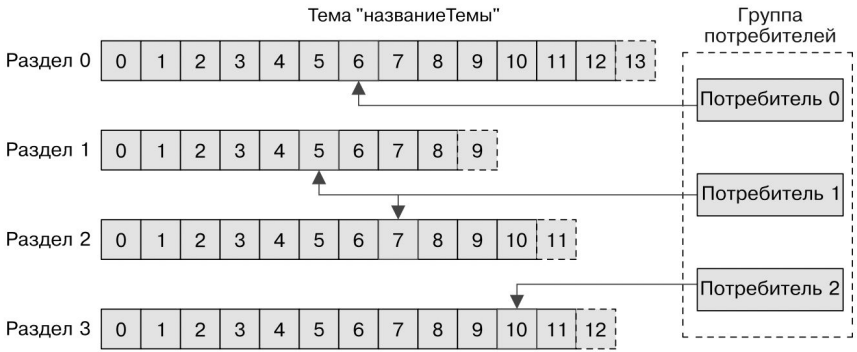
Как устроена Kafka?

- Consumer может читать из нескольких партиций.



Как устроена Kafka?

- Consumer может читать из нескольких партиций.
- **НО!** Два consumers из одной consumer group не могут читать одну партицию.



Kafka не резиновая!

- У Kafka есть **retention policy** (политика сохранения сообщений).

Kafka не резиновая!

- У Kafka есть **retention policy** (политика сохранения сообщений).
- *log.retention.bytes* или *log.retention.ms*

Факторы выбора формата данных

- Тип нагрузки на данные (чтение или запись)

Факторы выбора формата данных

- Тип нагрузки на данные (чтение или запись)
- Возможность сжатия данных (snappy, lz4, gzip)

Факторы выбора формата данных

- Тип нагрузки на данные (чтение или запись)
- Возможность сжатия данных (snappy, lz4, gzip)
- Расщепление файла на кусочки

Факторы выбора формата данных

- Тип нагрузки на данные (чтение или запись)
- Возможность сжатия данных (snappy, lz4, gzip)
- Расщепление файла на кусочки
- Эволюция схемы (schema evolution)

JSON формат

- Неразделимый формат (нельзя расщепить на кусочки)

JSON формат

- Неразделимый формат (нельзя расщепить на кусочки)
- Строчный формат

JSON формат

- Неразделимый формат (нельзя расщепить на кусочки)
- Строчный формат
- Схема интегрирована с данными

Формат данных Avro

- Данные хранятся в бинарном формате

Формат данных Avro

- Данные хранятся в бинарном формате
- Строковый формат хранения данных на диске

Формат данных Avro

- Данные хранятся в бинарном формате
- Строковый формат хранения данных на диске
- Схема отдельно от данных в формате JSON

Формат данных Avro

- Данные хранятся в бинарном формате
- Строковый формат хранения данных на диске
- Схема отдельно от данных в формате JSON
- Поддержка schema evolution

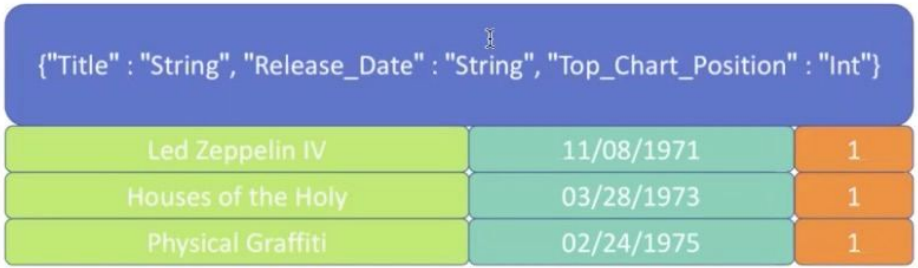
Формат данных Avro

- Данные хранятся в бинарном формате
- Строковый формат хранения данных на диске
- Схема отдельно от данных в формате JSON
- Поддержка schema evolution
- Расщепление файлов

Формат данных Avro

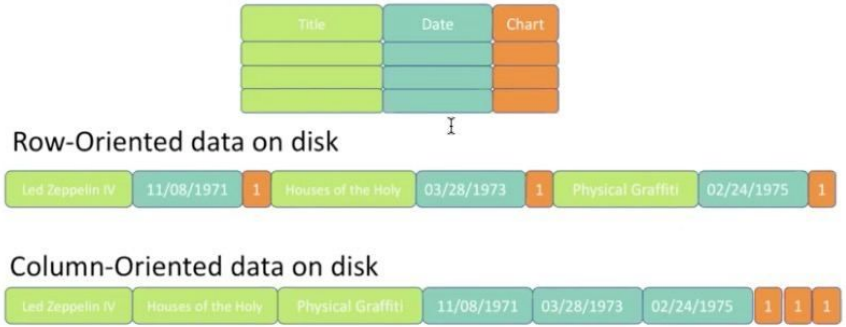
- Данные хранятся в бинарном формате
- Строковый формат хранения данных на диске
- Схема отдельно от данных в формате JSON
- Поддержка schema evolution
- Расщепление файлов
- Сжатие

Формат данных Avro

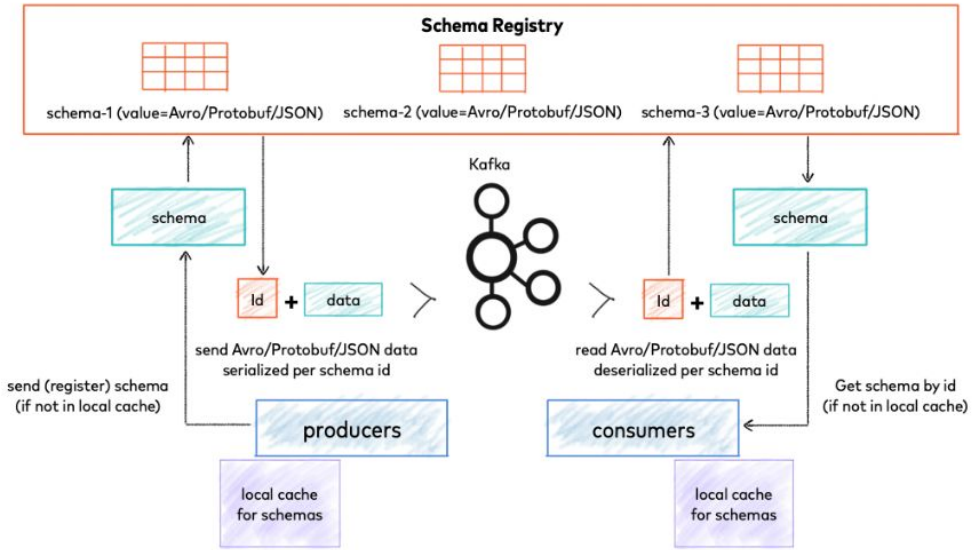


Формат данных Avro

Avro — строчный формат хранения данных на диске



Schema Registry



Confluent Schema Registry for storing and retrieving schemas

Что почитать

- https://towardsdatascience.com/10-common-software-architectural-patterns-in-a-nutshell-a0b47a1e9013
- https://book.huihoo.com/pdf/confluent-kafka-definitive-guide-complete.pdf
- https://highload.today/kolonochnye-bazy-dannykh/
- https://www.adaltas.com/en/2020/07/23/benchmark-study-of-different-file-format/
- https://towardsdatascience.com/big-data-file-formats-explained-dfaabe9e8b33