

Kafka - один из брокеров сообщений, построенный на модели Log (журнал).

Ранее мы с вами рассматривали модели pub-sub и очереди. Модель журнала не сильно отличается от pub-sub:

- Основной акцент на сохранении последовательности сообщений, как в журнале
- Сообщения записываются в журнал и могут быть прочитаны в том же порядке, в котором они были записаны
- Сообщения хранятся согласно настройкам, может быть задан длительный период хранения
- Сообщения могут читать разные подписчики, между брокером и консьюмером используется модель pull

Исходя из этого, сразу отметим когда используется Kafka - обработка больших объемов данных в реальном времени, где преимущественно важна последовательность и сохранность данных.

Некоторые примеры, в каком применении Kafka точно будет полезна:

- Обработка большого количества событий от разных источников
- Системы логирования и журналирования
- Сбор метрик
- Межсервисное взаимодействие с использованием больших объемов данных
- Обработка и передача большого количества небольших сообщений в реальном времени
- Хранилище информации из разных источников

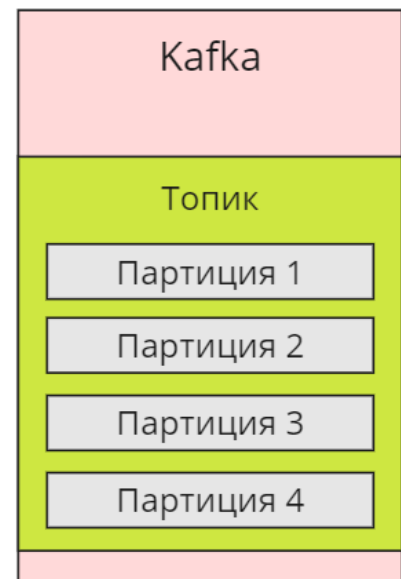
Kafka сейчас из самых популярных брокеров сообщений и используется почти в каждой крупной IT-организации.

Начнем изучение с терминов.

- Партиция - место (лог) хранения сообщений в Kafka. Сообщения строго помещаются в одну из партиций. Партиция также обеспечивает строгий порядок сообщений и обязательно находится в одном из топиков.
- Топик - группа, которая хранит в себе одну или несколько партиций. Топик в Kafka это аналогия топика в pub-sub и в нее должны поступать однотипные сообщения. Для разных бизнес-логик создают разные топики. Например, если у нас Kafka используется для аналитики веб-сайта, то можно иметь топики для различных событий, таких как "посещение страницы", "регистрация пользователя" и "оформление заказа".
- Оффсет - коротко обсуждали, что это такое. В Kafka это "указатель" в партиции, он обозначает место, где остановился тот или иной консьюмер. Консьюмер может запросить сообщения с определенного оффсета. Если оффсеты хранятся в Kafka, то они содержатся в системном топике "consumer_offsets".
- Консьюмер группа - группа консьюмеров с общим оффсетом для всех партиций определенного топика.

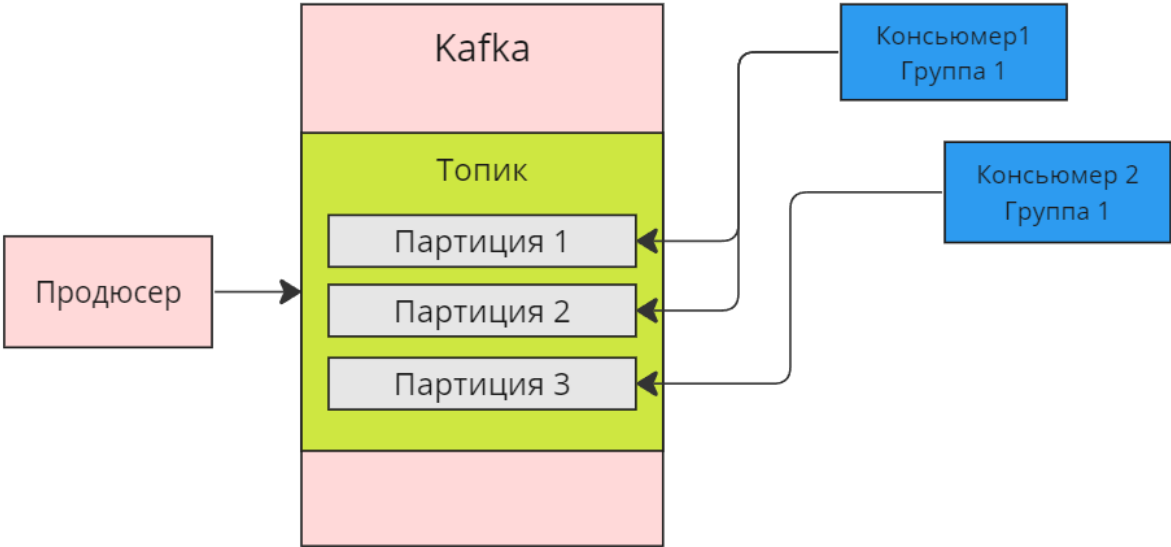
Консьюмер при подключении к Kafka знает, какой топик нужно читать и иногда знает из какой именно партиции.

Наглядная схема, из чего состоит Kafka:

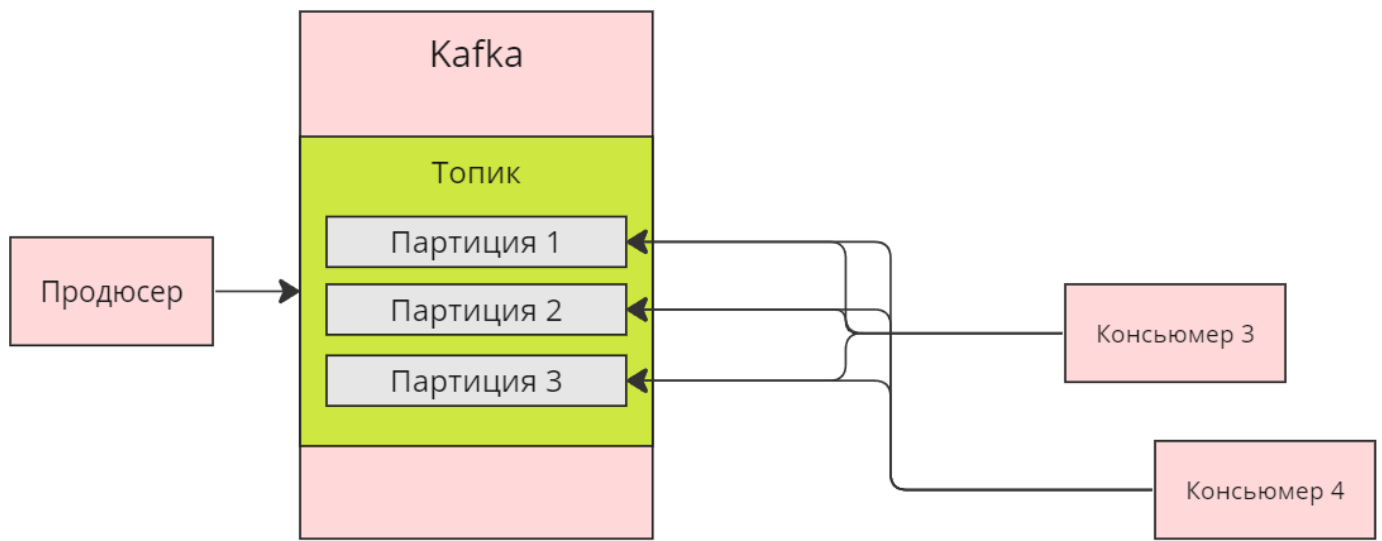


Топиков как и партиций, может быть много. Запомните - в брокере есть топики, в топиках партиции, а в партиции хранятся сообщения с определенным оффсетом (позицией).

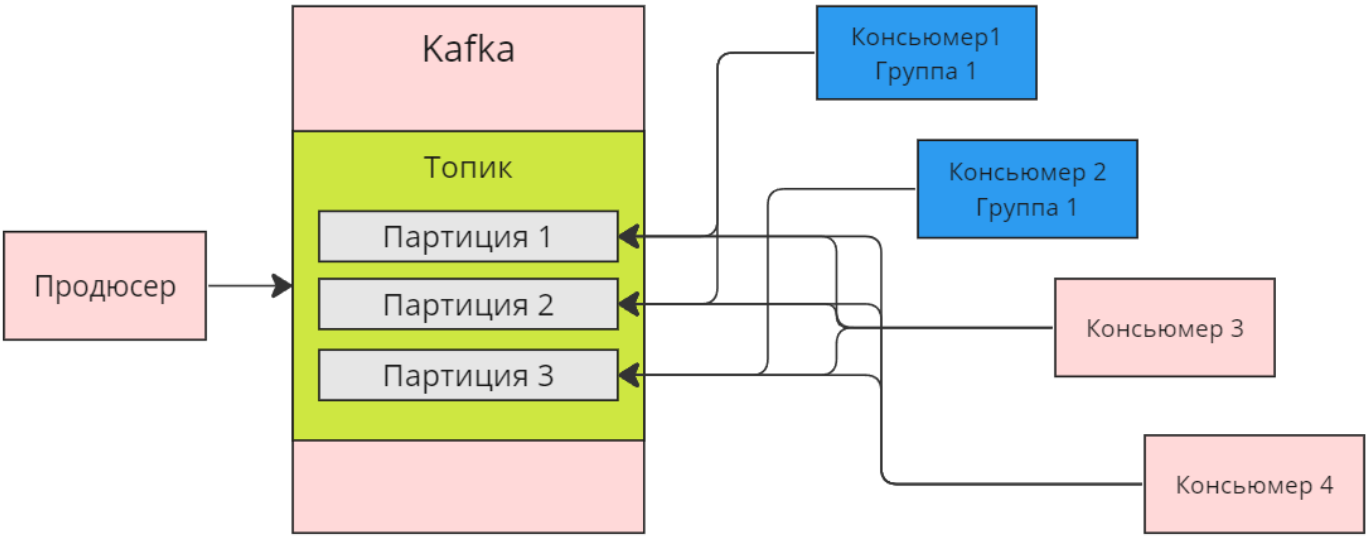
Давайте представим, что у нас есть один продюсер, несколько консьюмеров и консьюмер группа. А в Kafka есть один топик, с тремя партициями. Взаимодействие с консьюмер группой будет выглядеть так:



Взаимодействие с консьюмерами будет выглядеть так:



Объединенная схема взаимодействия так:



Разберем по шагам, как происходит взаимодействие:

1. Продюсер передает сообщение. Он передает сообщение именно в топик, и Kafka самостоятельно подбирает партицию (один из механизмов - Round robin). Хотя продюсер может отправить запрос, указав определенную партицию.
2. Kafka получает и записывает сообщение в партицию топика, партицию выбирает самостоятельно или записывает в нужную согласно запросу продюсера.
3. Kafka никуда не отправляет сообщение, используется модель pull. Сообщения хранятся согласно настройкам брокера. Кстати, можно и удалять после прочтения консьюмером, так работают обычно модели очередей.
4. Консьюмеры подключаются к топiku и начинают читать сообщение. Каждый консьюмер может прочитать каждое сообщение в любой партиции. То есть как и в pub-sub, каждый может прочитать одно и тоже сообщение, множество раз. Ограничений нет. В данном случае мы подключили консьюмеры ко всем партициям.
5. Группа консьюмеров подключается к топiku и начинают читать сообщения. Здесь уже есть ограничение на чтение партиций. В своей группе один консьюмер может прочитать сообщение только в партиции, которую не читает другой консьюмер в группе. Согласно схеме, консьюмер 1 читает партицию один и два. Консьюмеру 2 остается возможность читать только третью партицию.

Давайте закрепим теорию и продолжим погружение.