

# Ментальная модель Kafka

Базовые понятия Apache Kafka  
для разработчиков, QA и DevOps-  
инженеров

# Содержание

1. Сценарии использования
2. Что такое консумер, продюсер и брокер
3. Как связаны топики, партиции и сегменты
4. Формат сообщения Kafka
5. Лидер партиции и репликация данных
6. Партицирование
7. Гарантии доставки сообщений
8. Идемпотентность
9. Что такое консумер-группа
10. Ребалансировка консумеров

# Сервисы управляют объектами

Динамические лимиты

Адреса доставки

Операционные зоны

Координаты партнёров

Рекомендации



Маршрутные листы

Уведомления

Фискализация

Сборки

Смены

# Сервисы

## СОЗДАЮТ СОБЫТИЯ

Установка лимита

Смена операционной зоны

Добавление рекомендации

Отправка уведомления

Назначение сборки

Изменение адреса доставки

Обновление координат

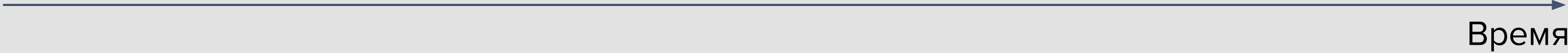
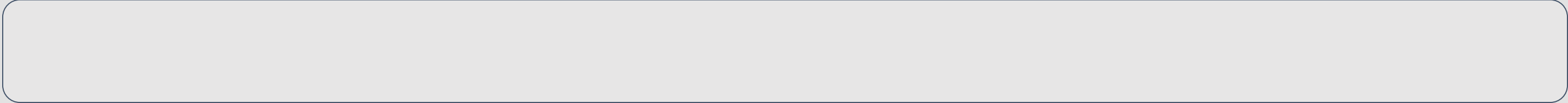
Создание маршрутного листа

Подготовка фискального чека

Закрытие смены



# Поток событий



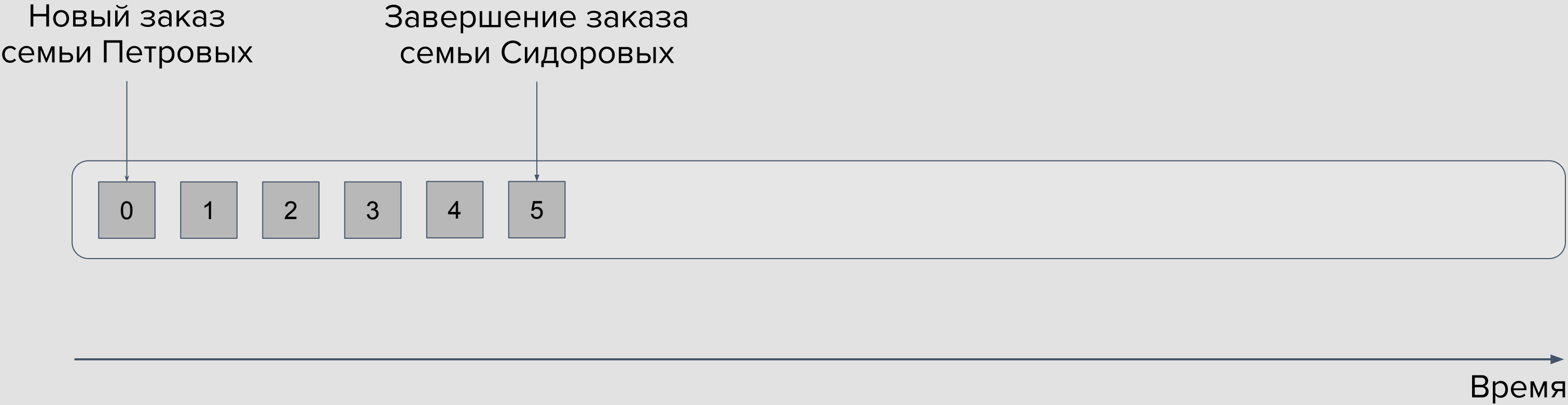
# Поток событий как **ЛОГ**

Новый заказ  
семьи Петровых

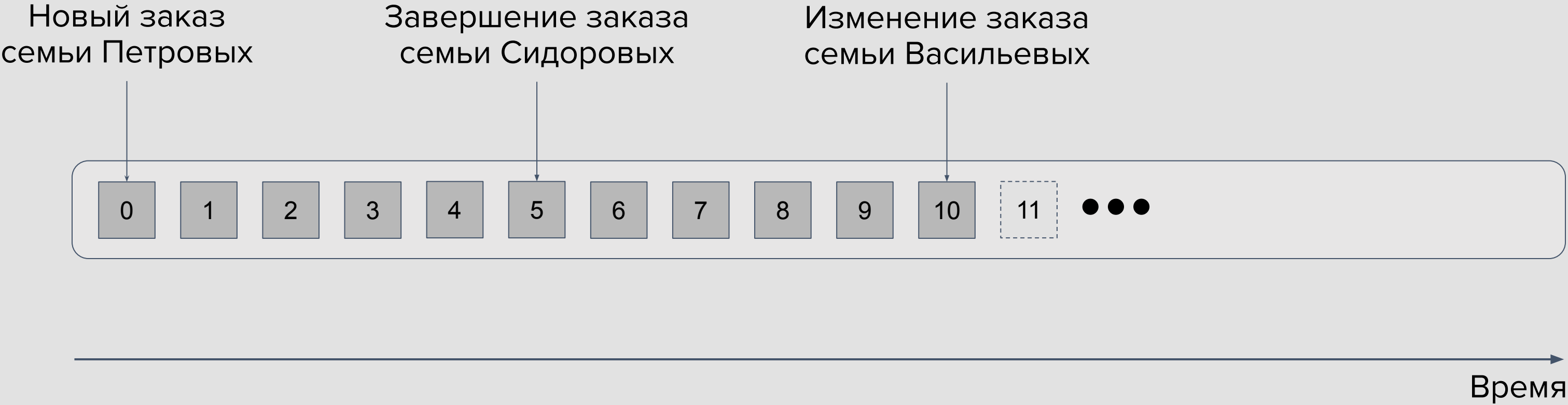
0

Время

# Поток событий как **ЛОГ**



# Поток событий как **ЛОГ**



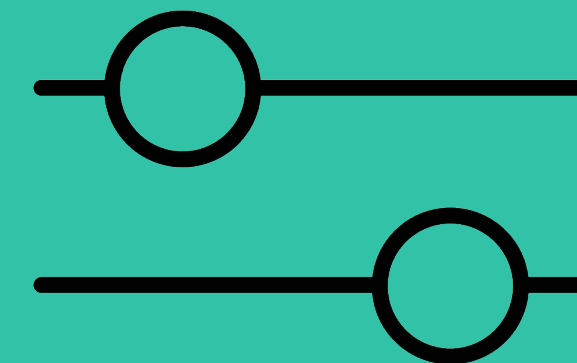




**Единая платформа** для  
связи сервисов и событий

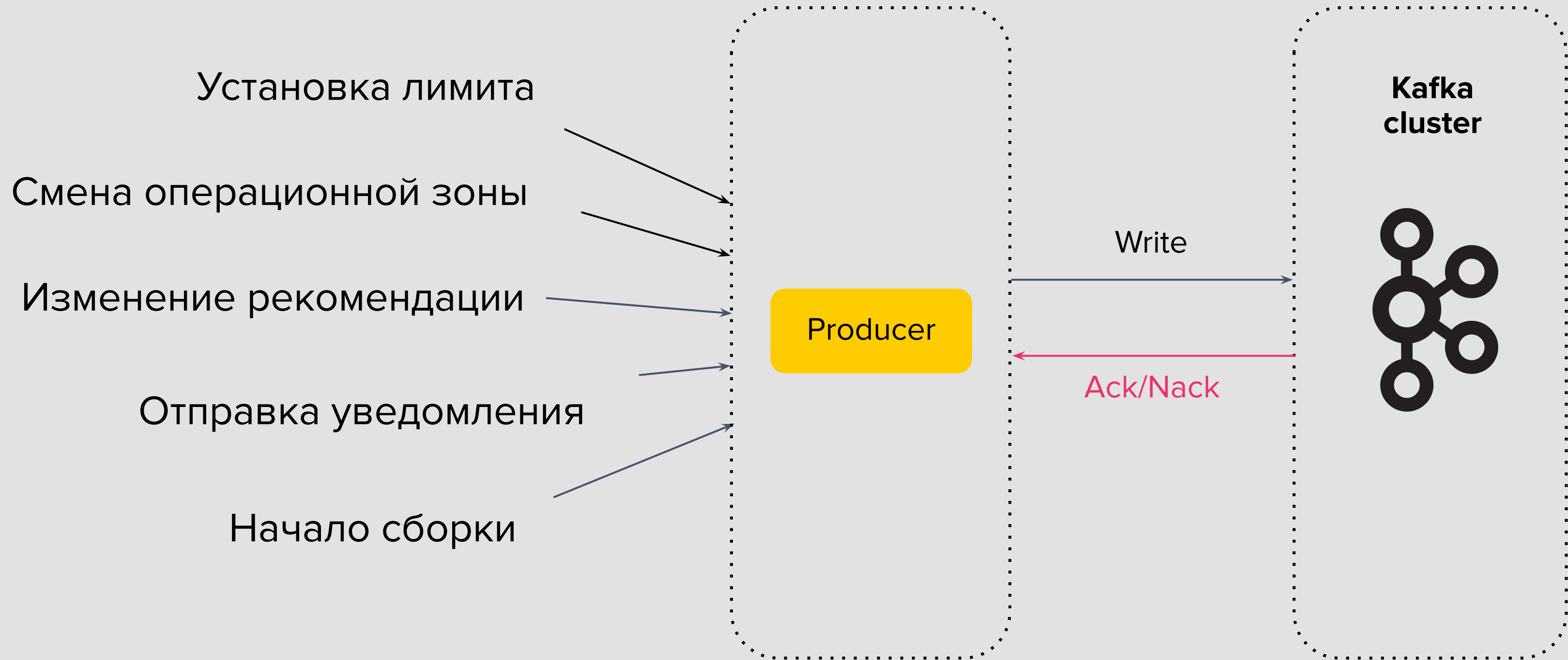


**Поток событий в  
реальном времени**

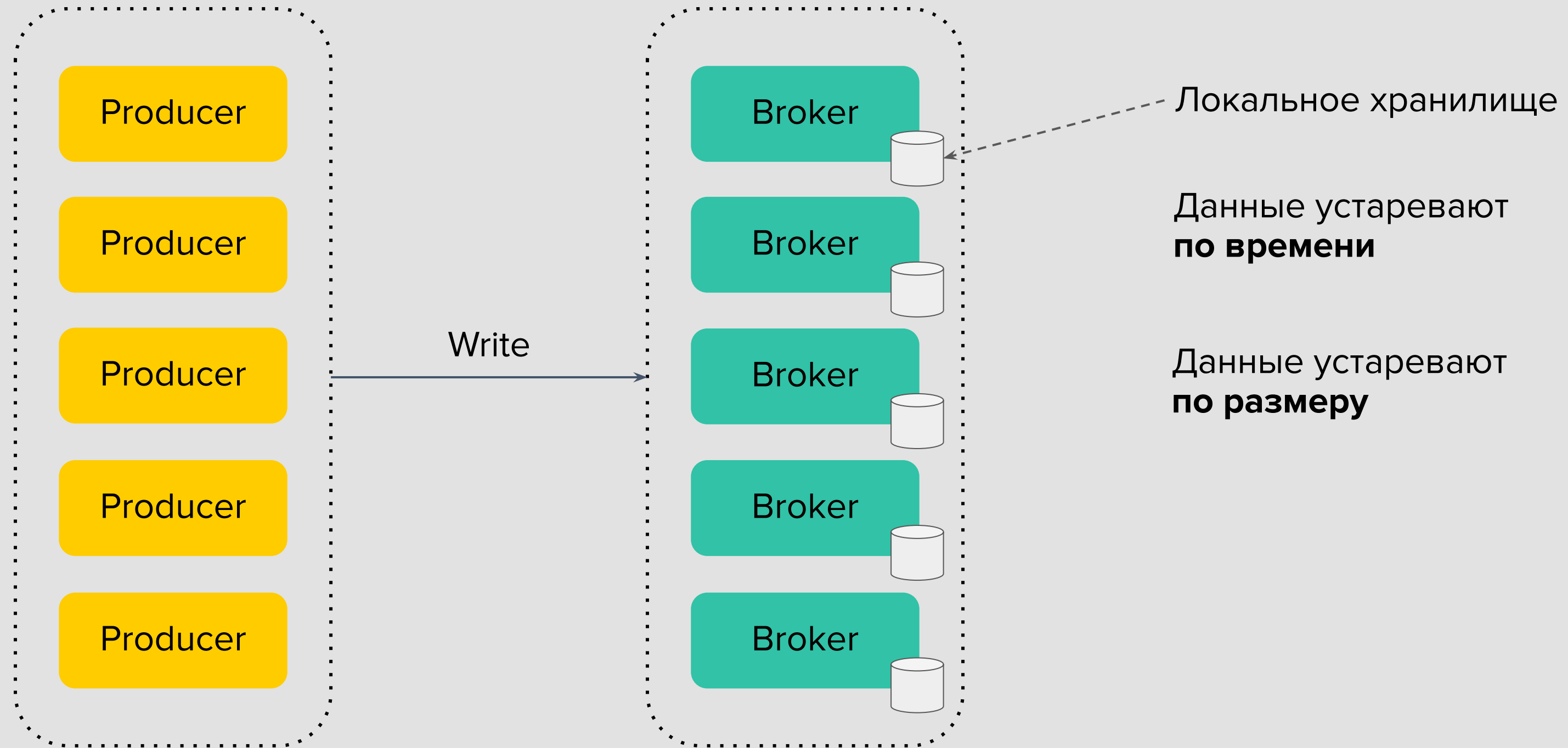


**События упорядочены  
во времени**

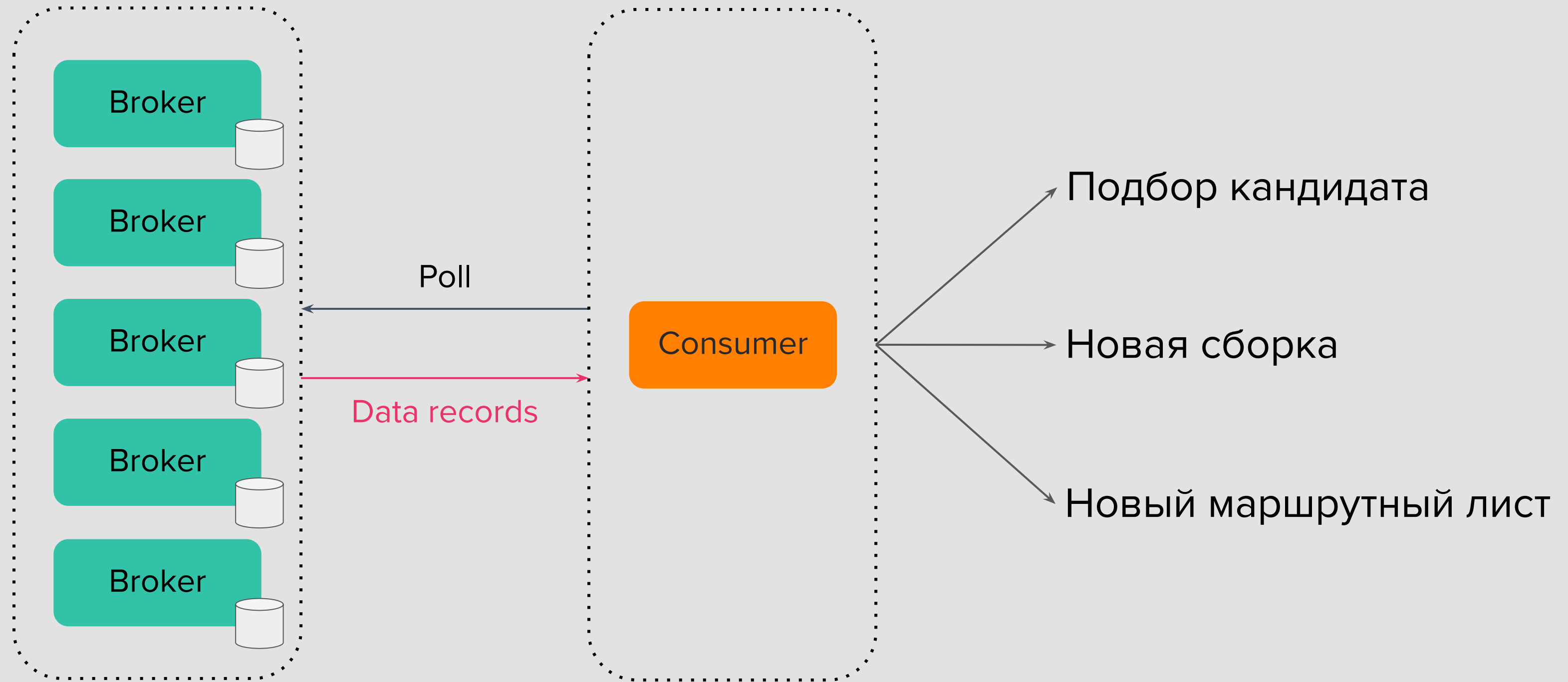
# Продюсеры



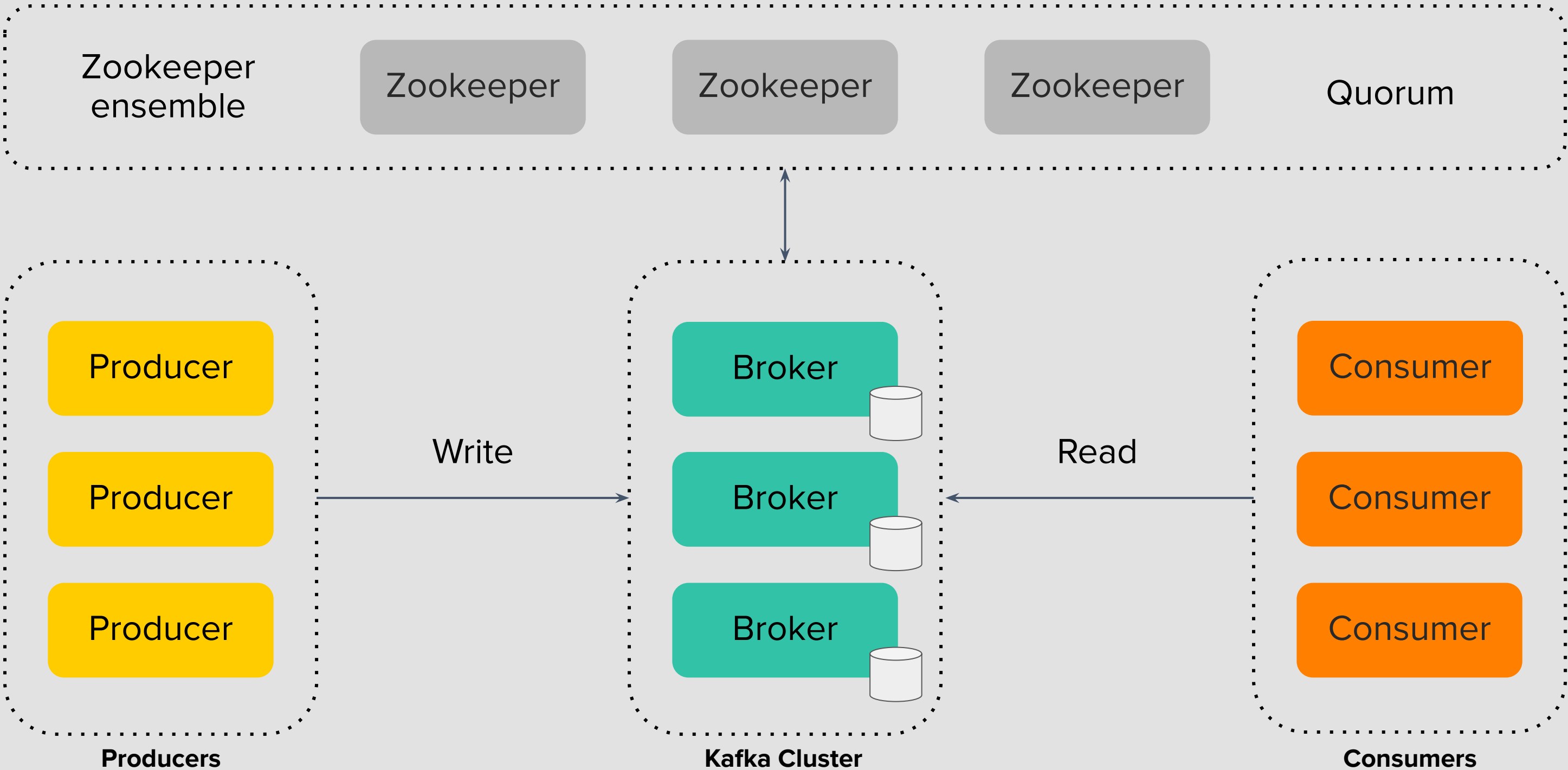
# Брокеры



# Консумеры

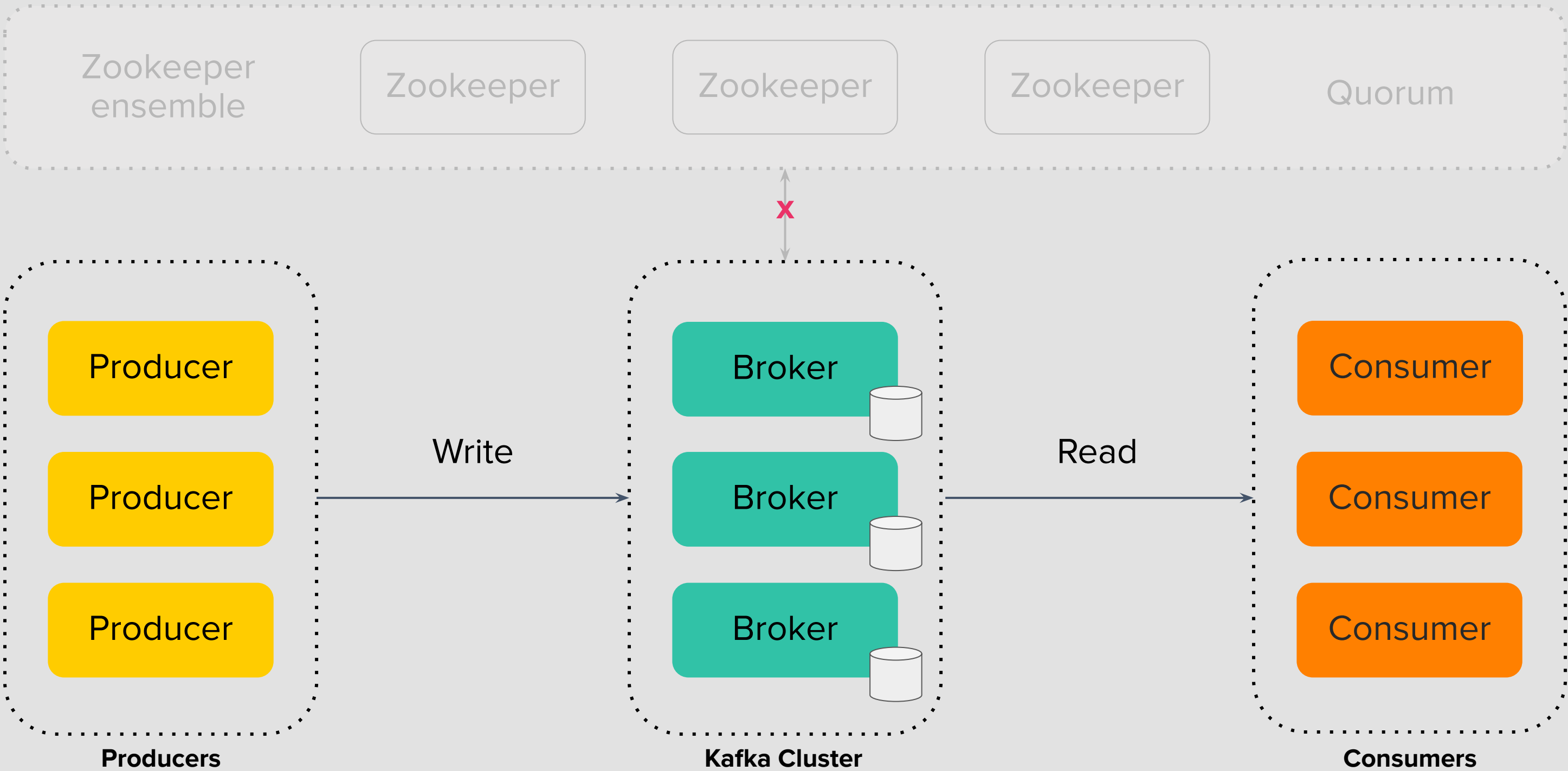


# Архитектура

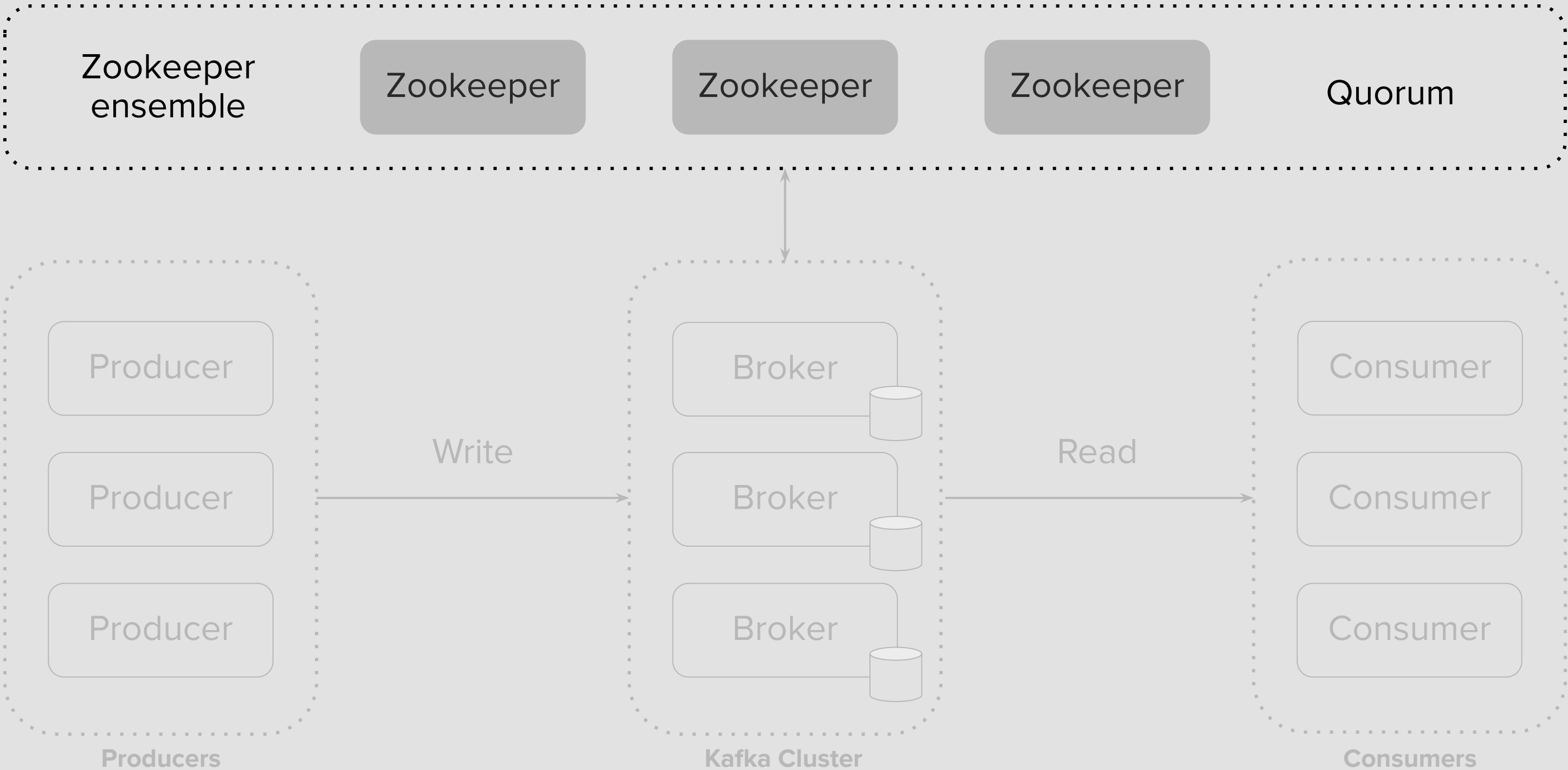


# Архитектура

## KRaft: Kafka Raft Metadata protocol



# Архитектура



# Архитектура



## 01

Управление  
кластером



# Архитектура



01

Управление  
кластером

02

Обнаружение сбоя и  
восстановление

# Архитектура



01

Управление  
кластером

02

Обнаружение сбоя и  
восстановление

03

Хранение ACL и  
секретов

# Изоляция консумеров и продюсеров

# 01

Консумеры и  
продюсеры разделены

# Изоляция консумеров и продюсеров

## 01

Консумеры и  
продюсеры разделены

## 02

Медленные консумеры  
не влияют на продюсеры

# Изоляция консумеров и продюсеров

01

Консумеры и  
продюсеры разделены

03

Добавление консумеров  
не влияет на продюсеры

02

Медленные консумеры  
не влияют на продюсеры

# Изоляция консумеров и продюсеров

01

Консумеры и  
продюсеры разделены

03

Добавление консумеров  
не влияет на продюсеры

02

Медленные консумеры  
не влияют на продюсеры

04

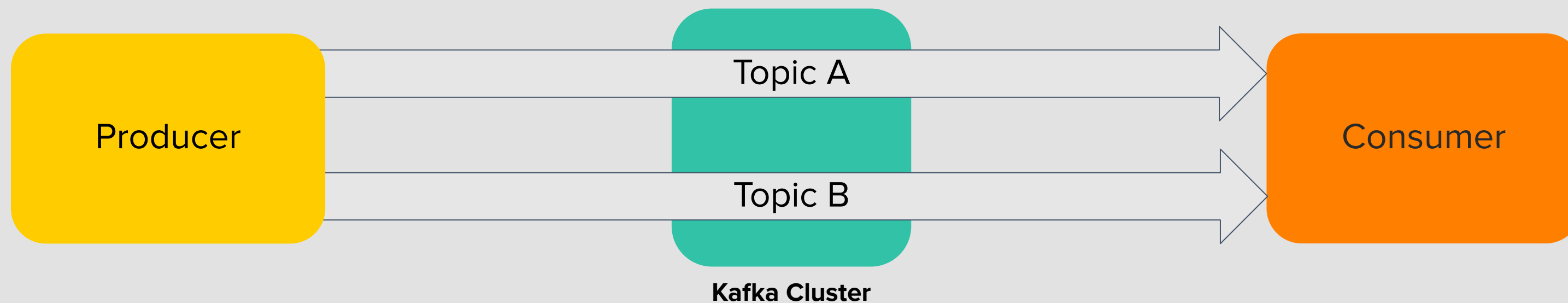
Сбойный консумер не  
влияет на работу

# Брокеры



# Топик

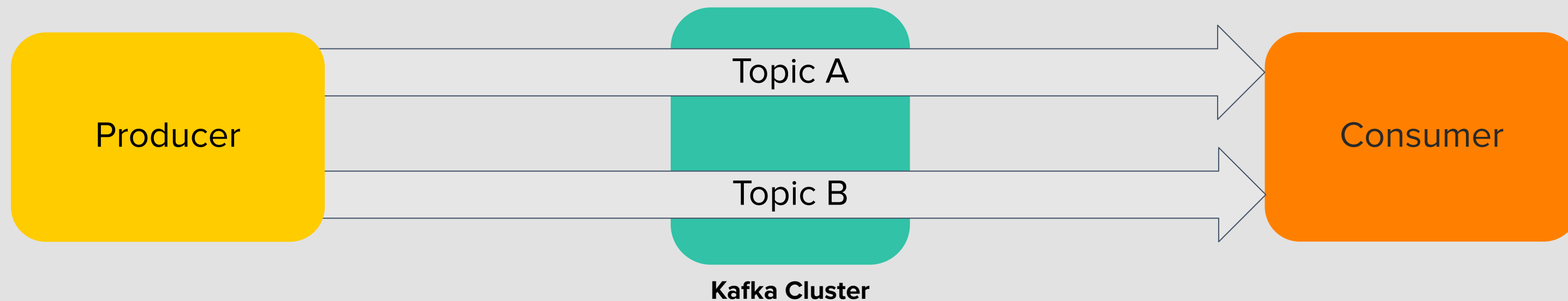
## Логическое представление категорий сообщений в группы



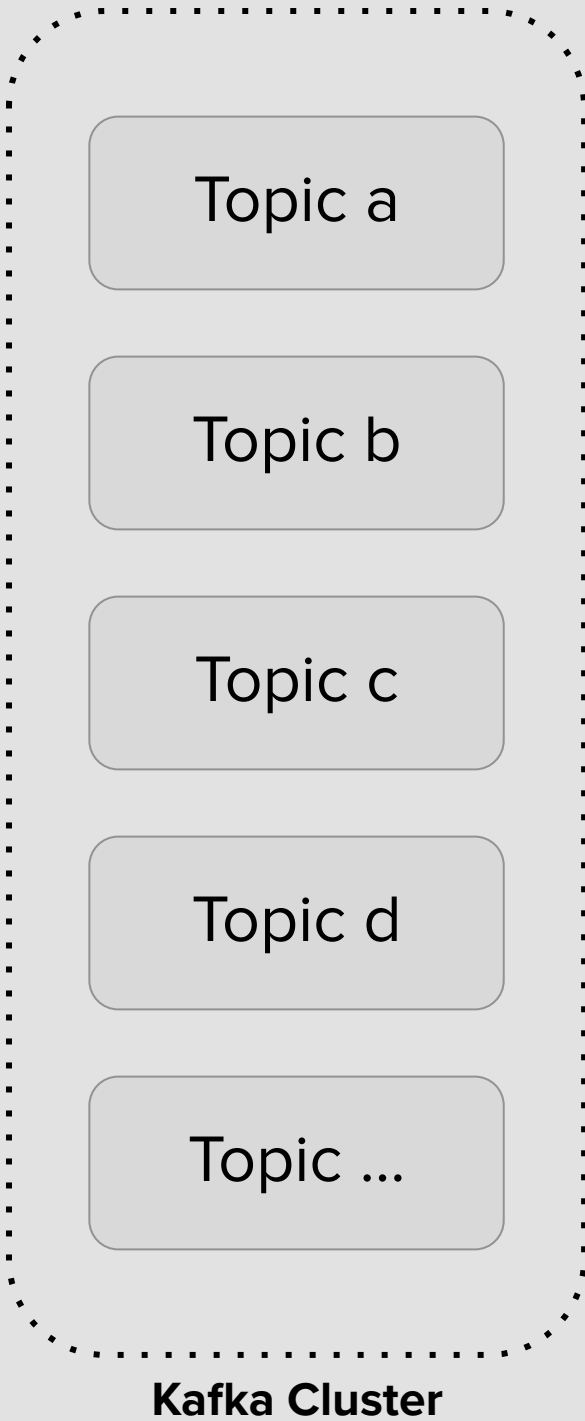


# Топик

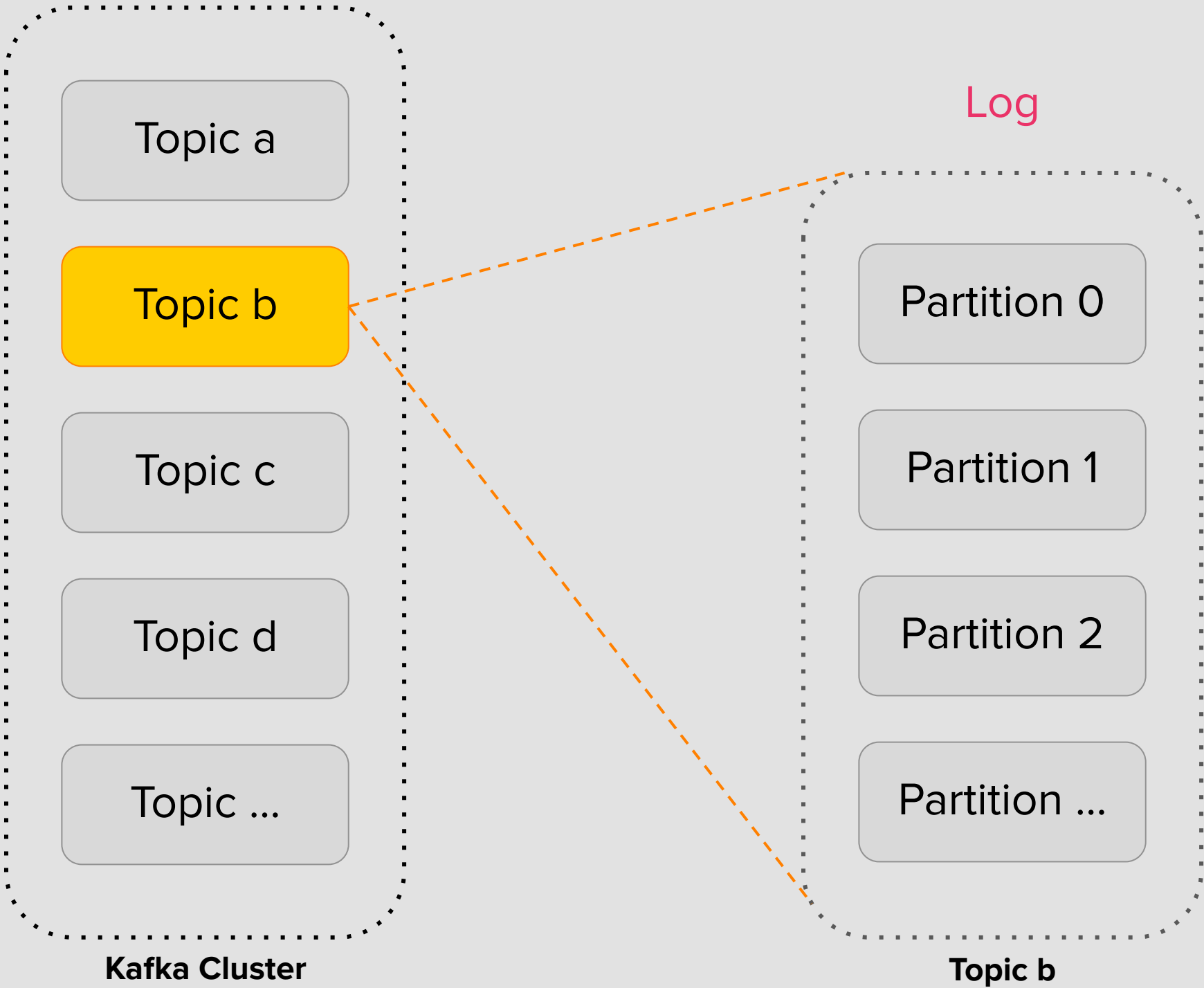
## Логическое представление категорий сообщений в группы



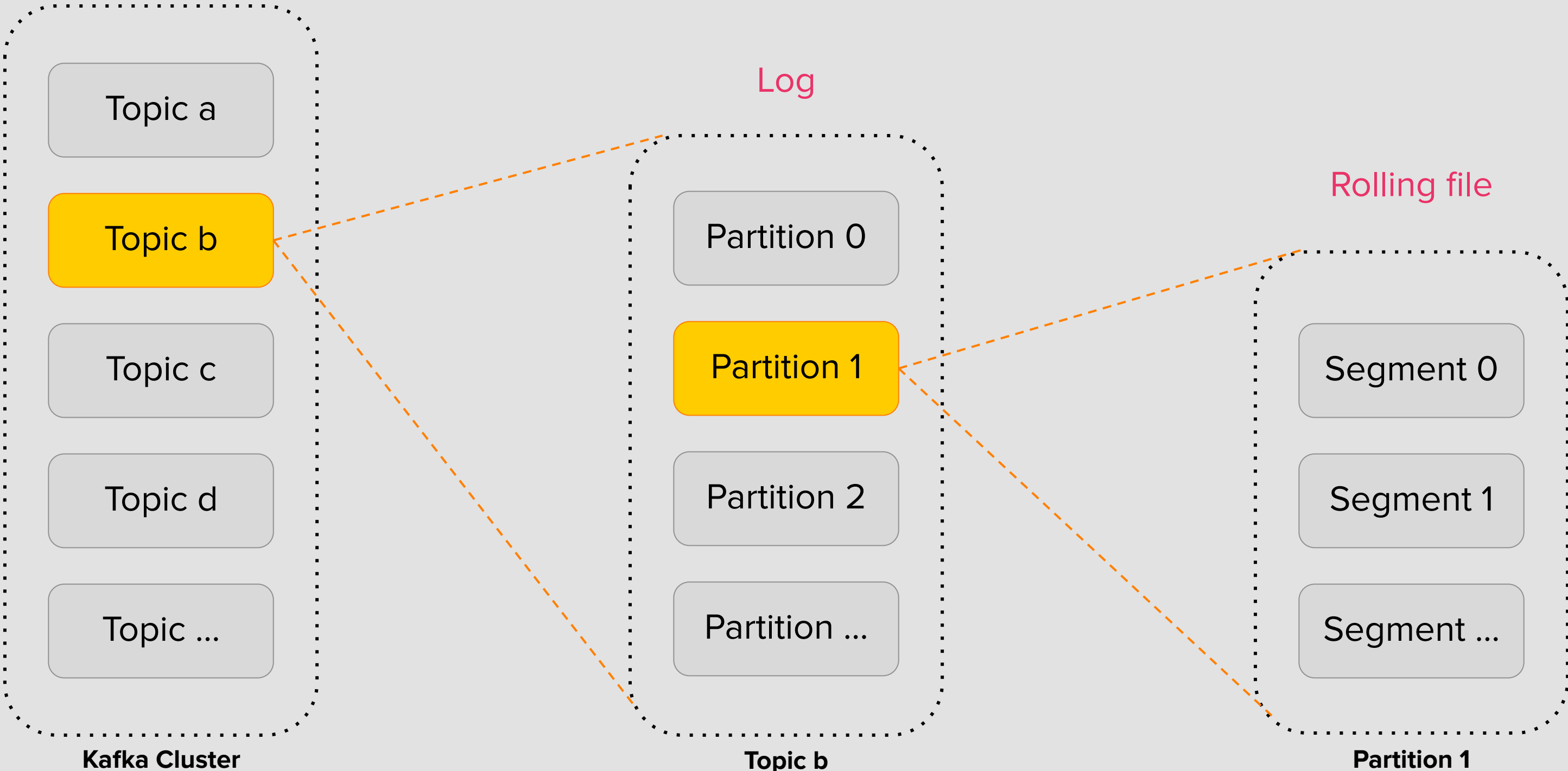
# Топики, партиции и сегменты



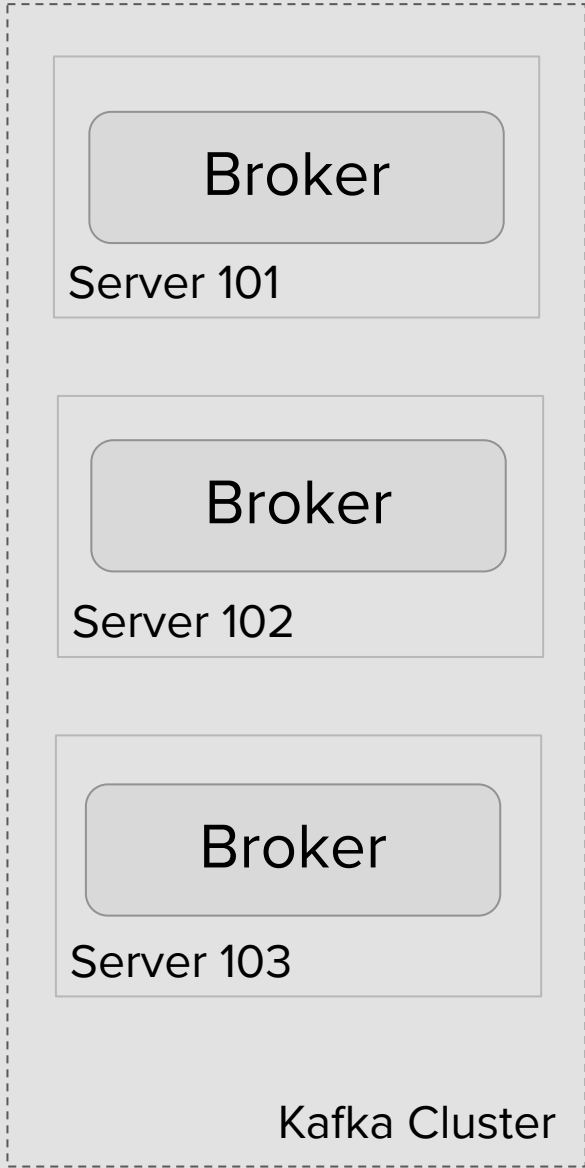
# Топики, партиции и сегменты



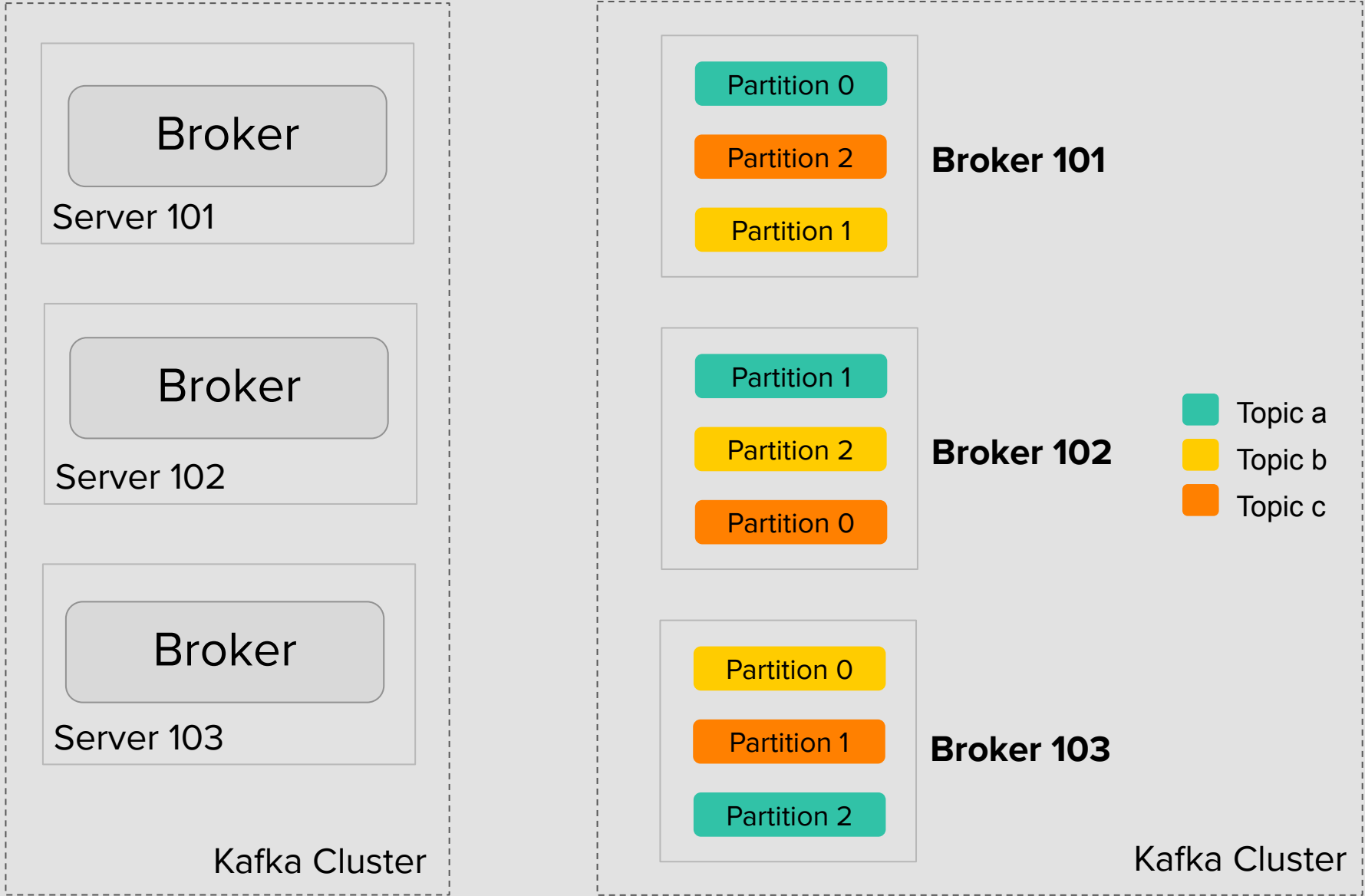
# Топики, партиции и сегменты



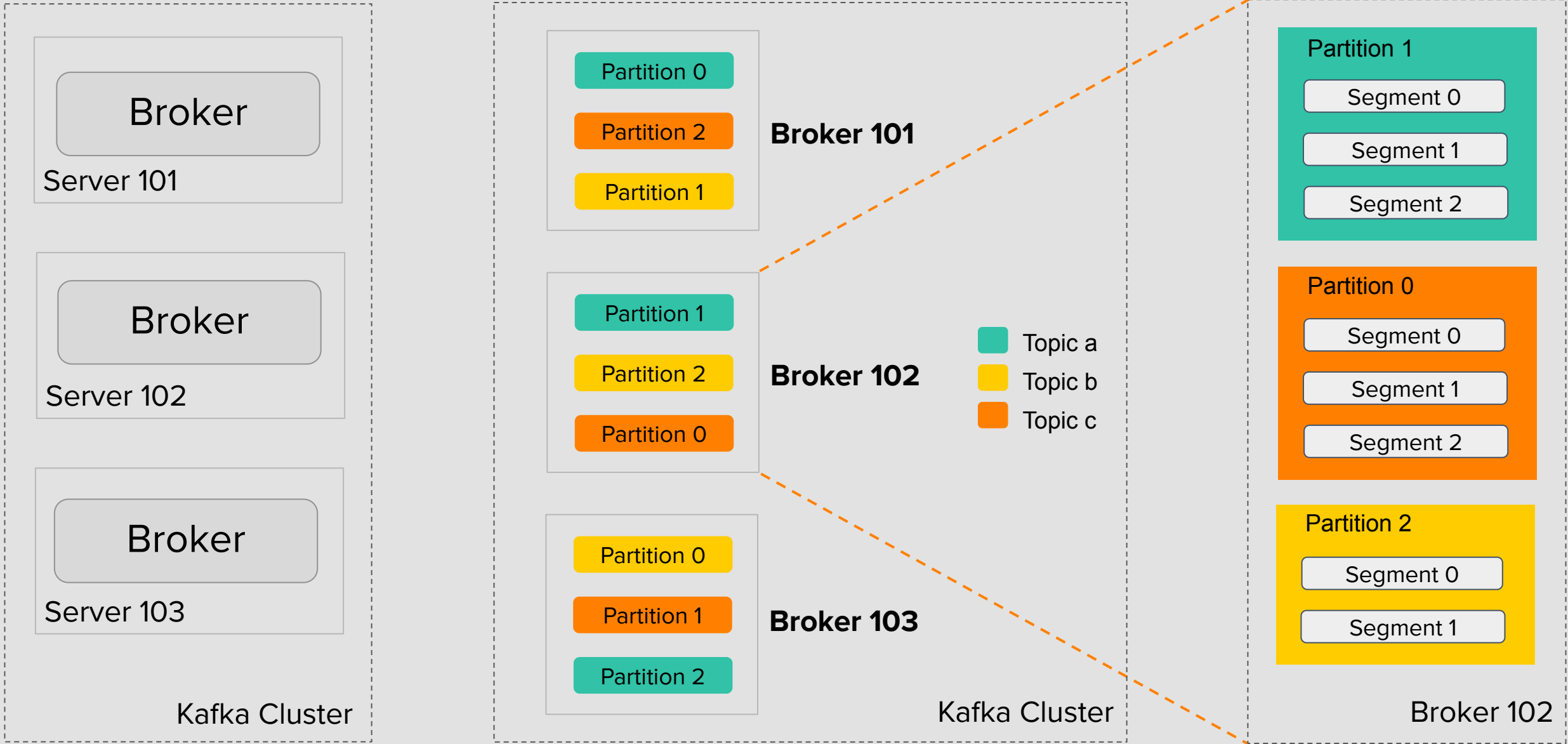
# Топики, партиции и сегменты



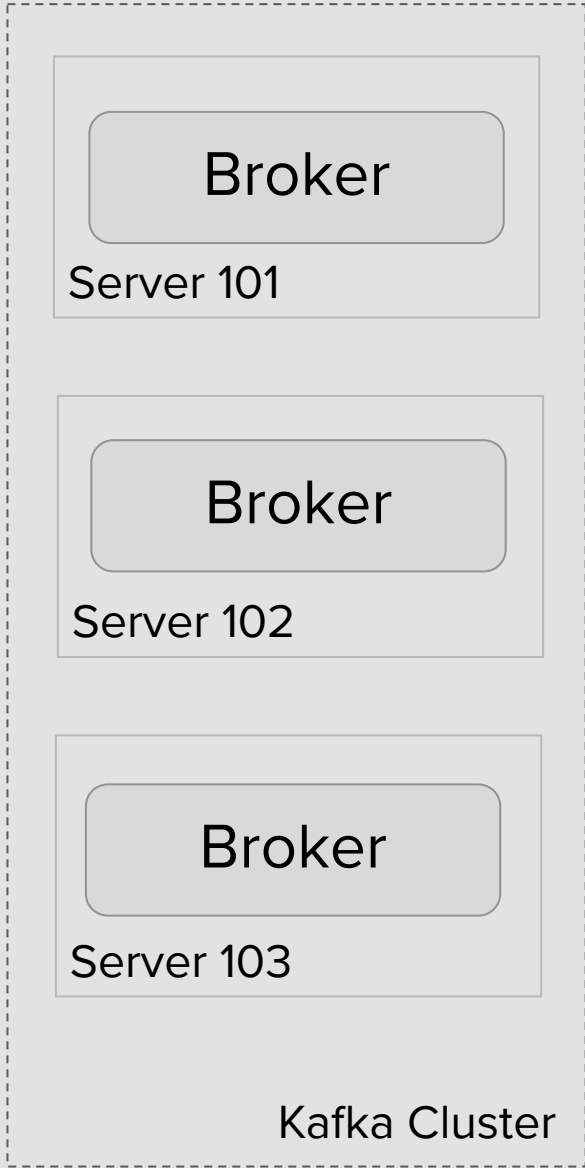
# Топики, партиции и сегменты



# Топики, партиции и сегменты



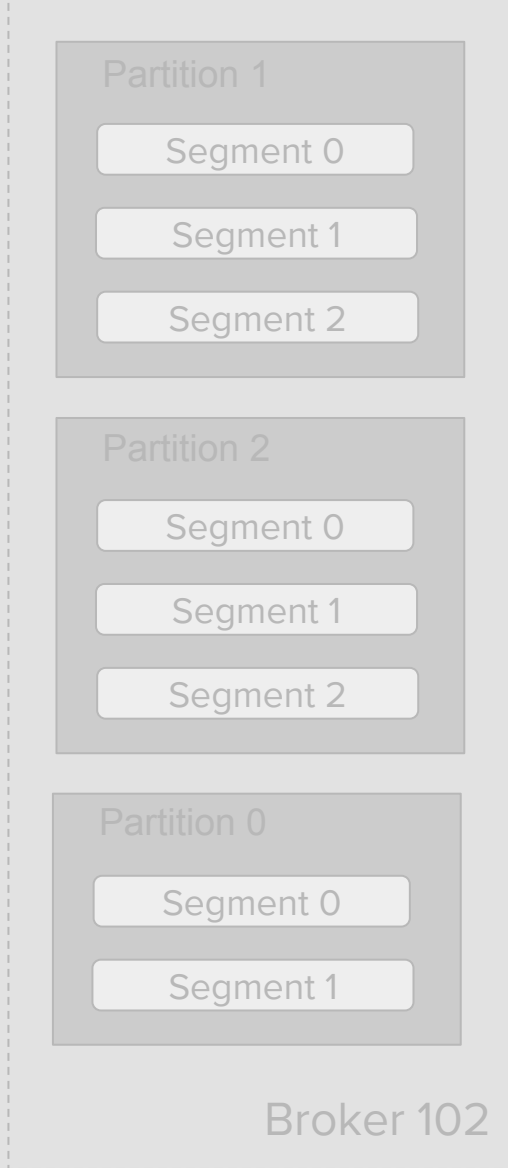
# Топики, партиции и сегменты



Масштабирование



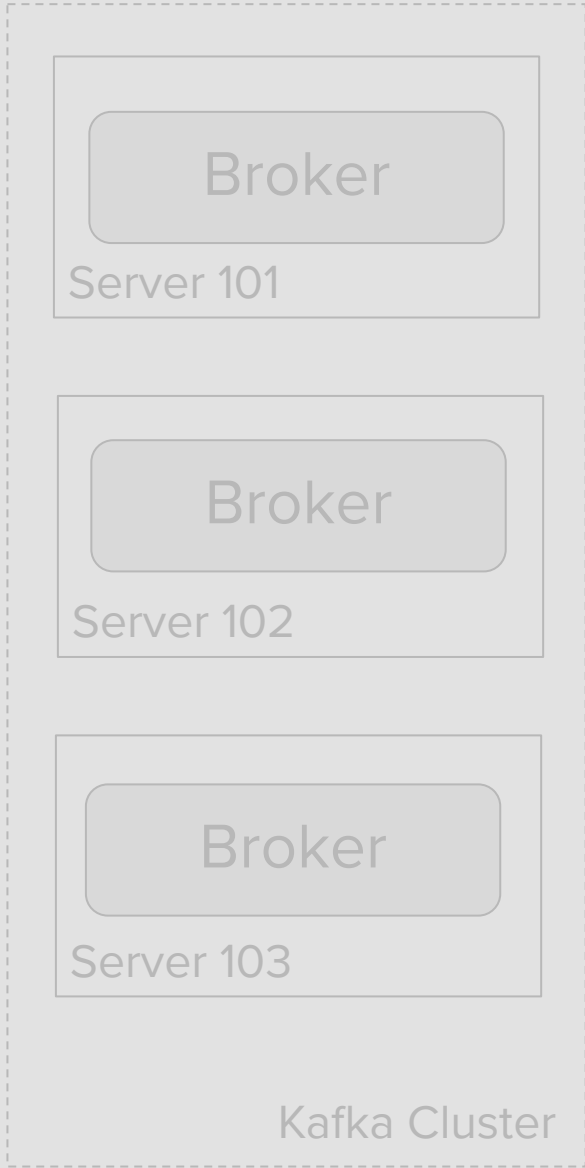
Параллелизм



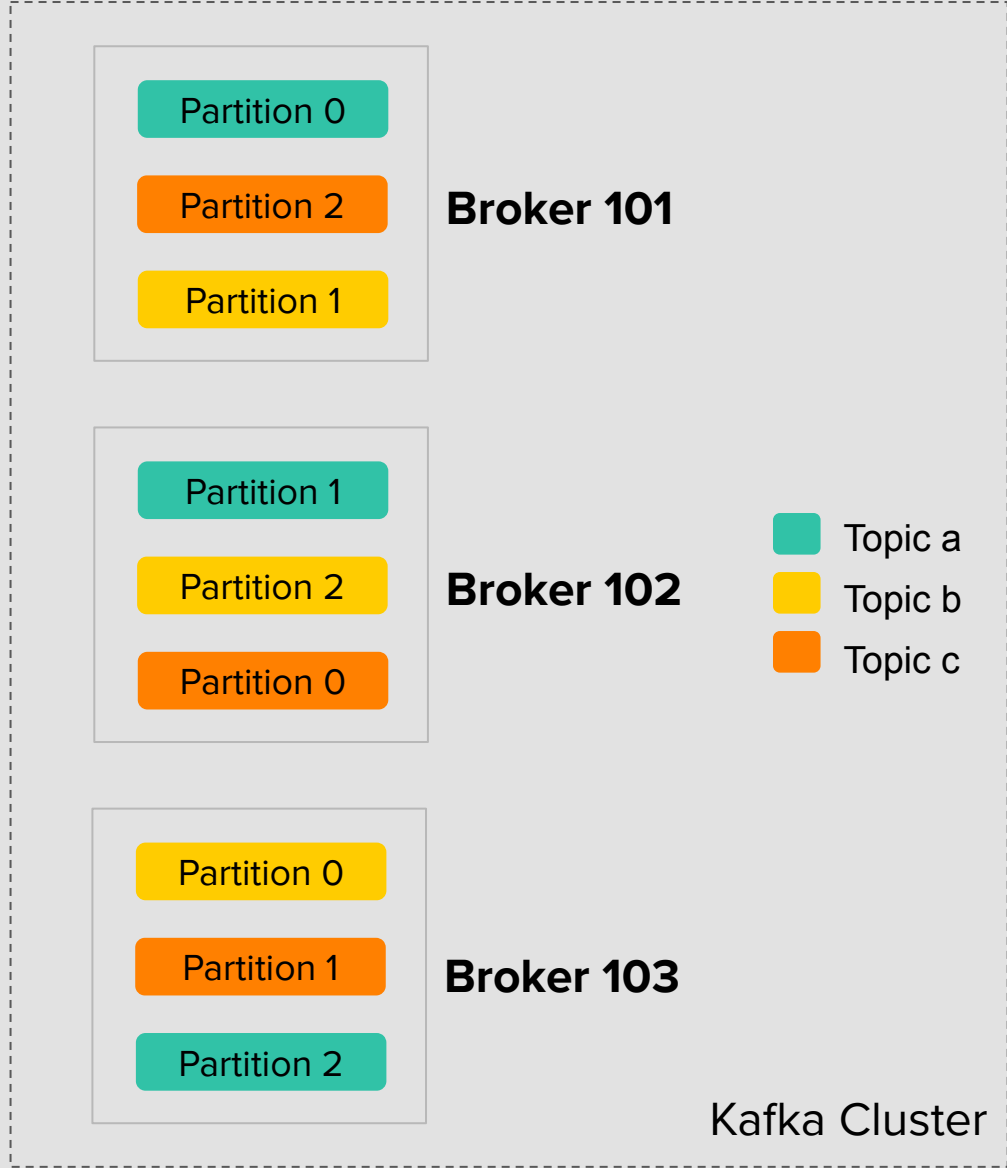
Ротируемые файлы



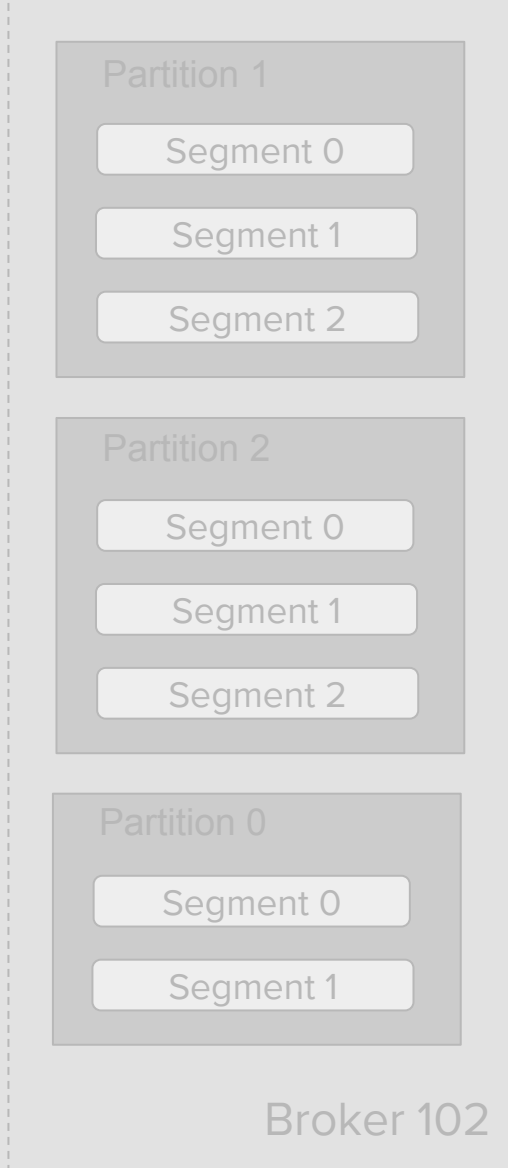
# Топики, партиции и сегменты



Масштабирование

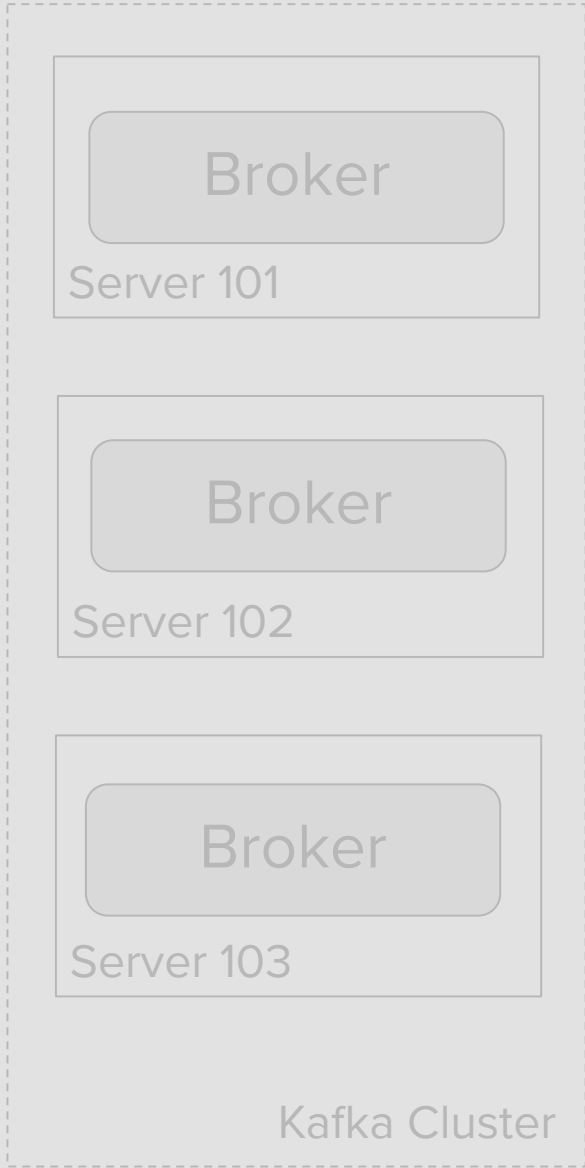


Параллелизм



Ротируемые файлы

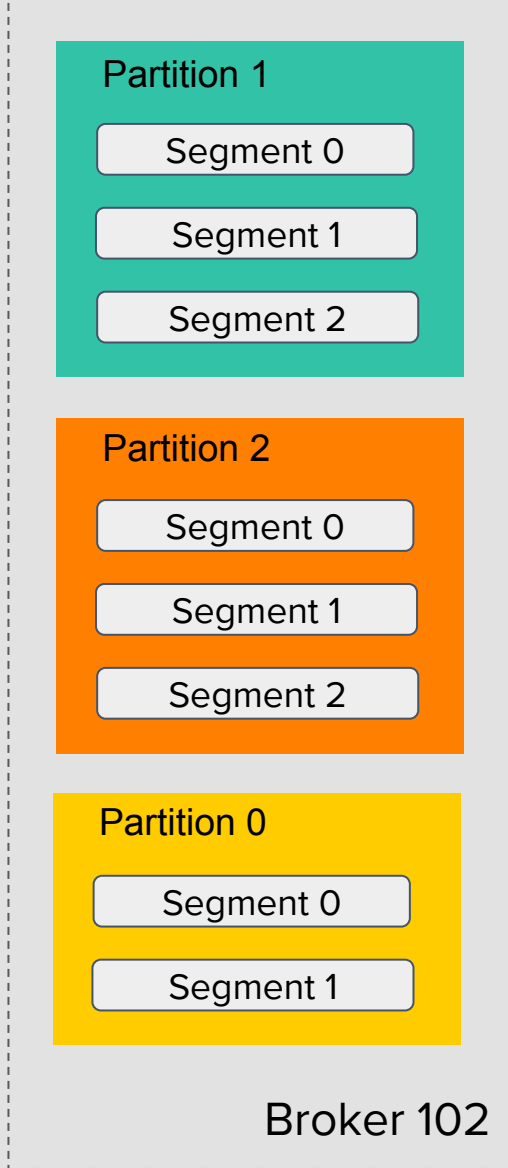
# Топики, партиции и сегменты



Масштабирование

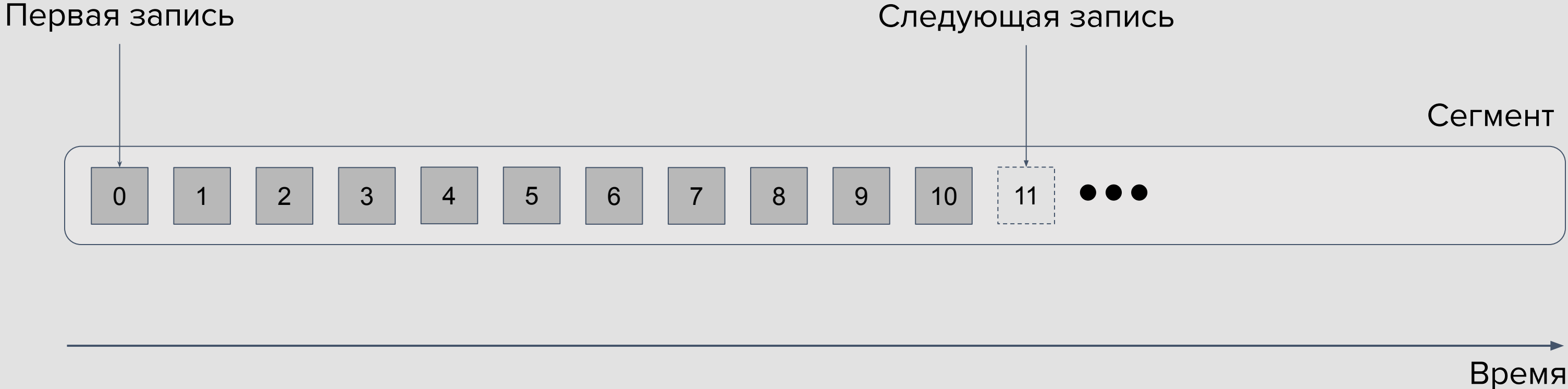


Параллелизм

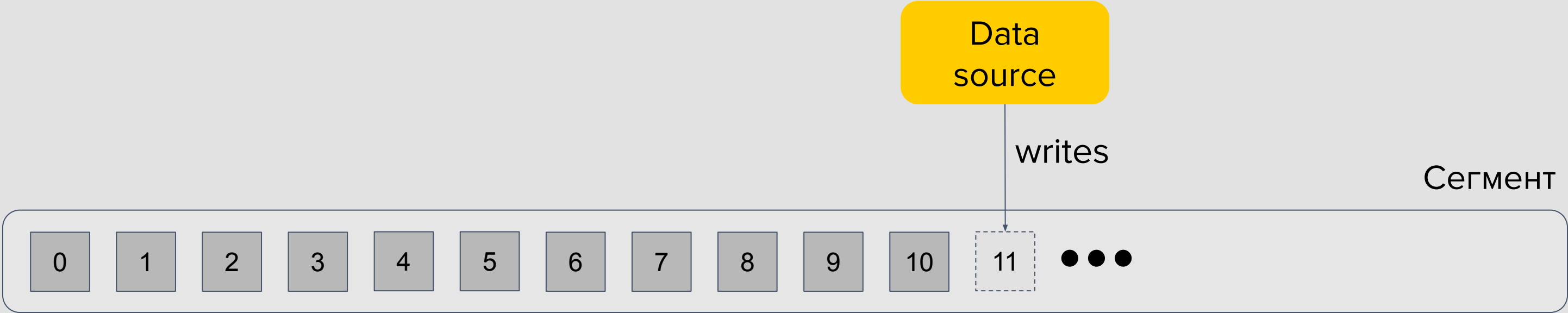


Ротируемые файлы

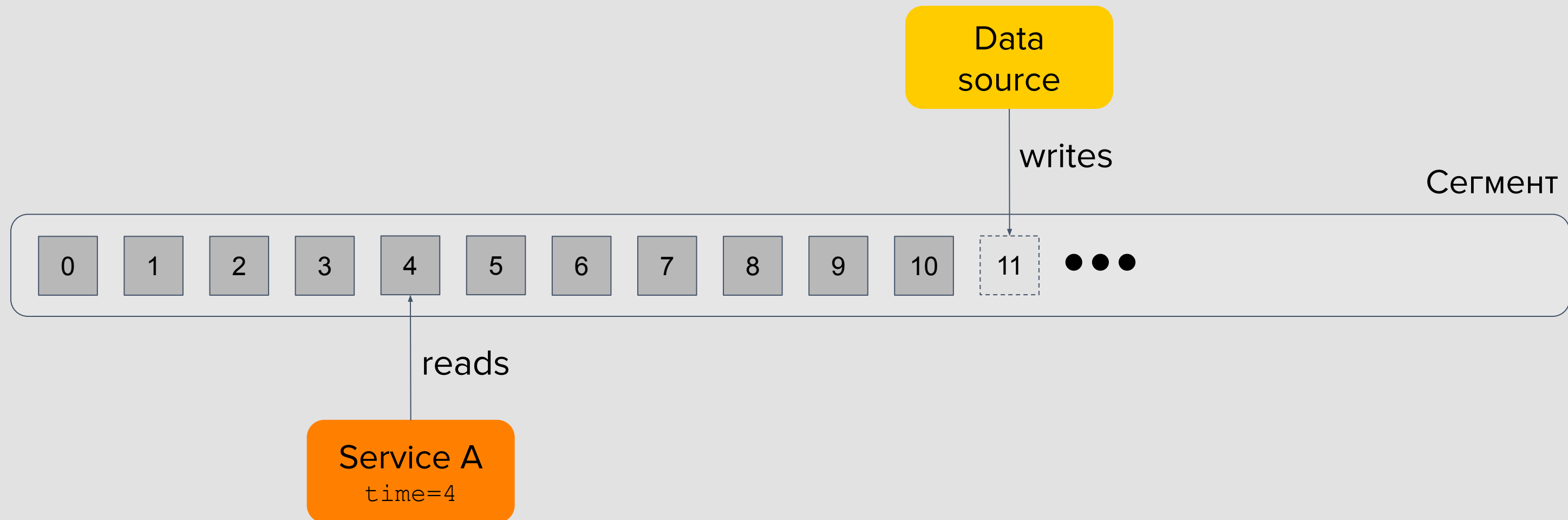
# Лог



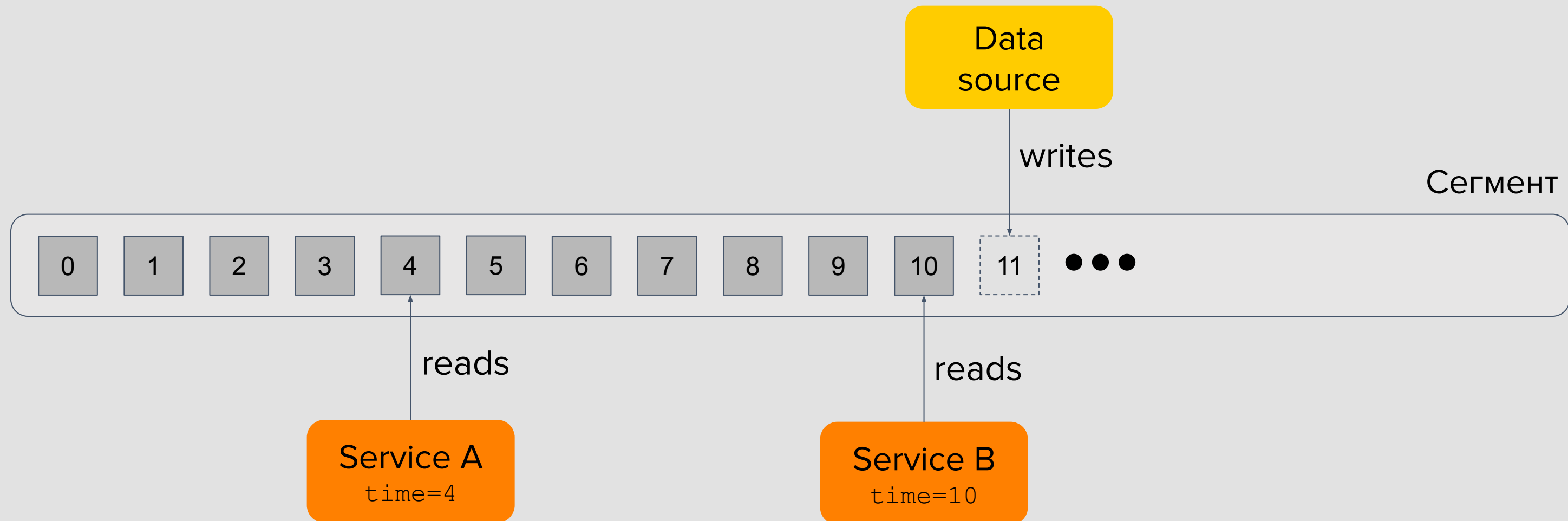
# Поток данных



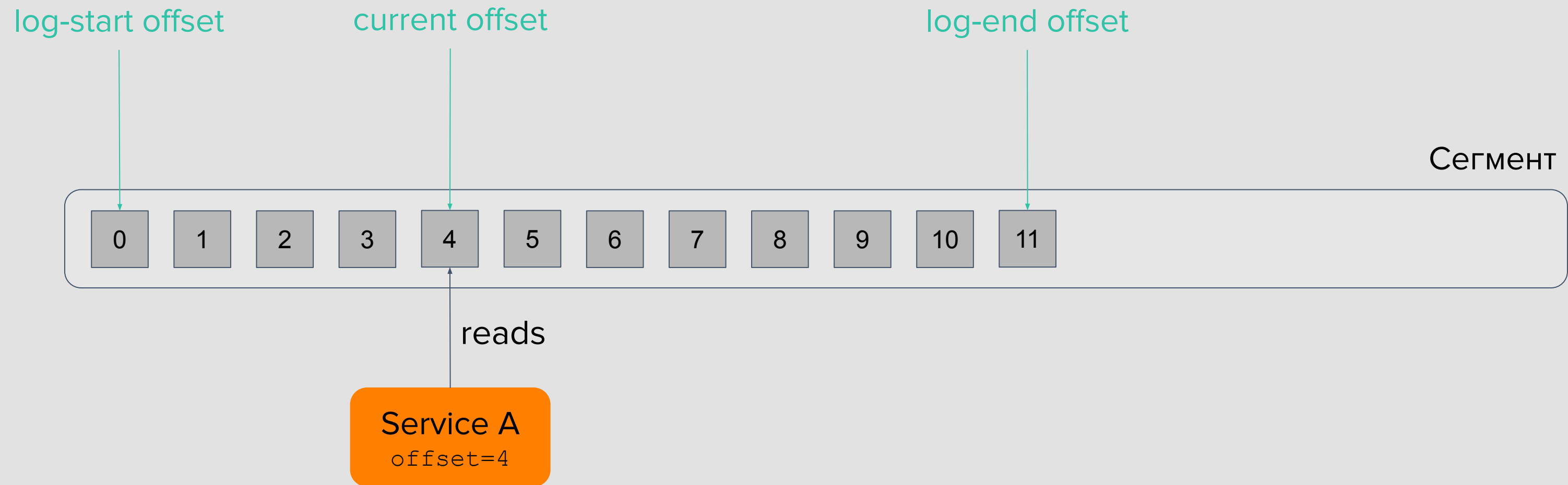
# Поток данных



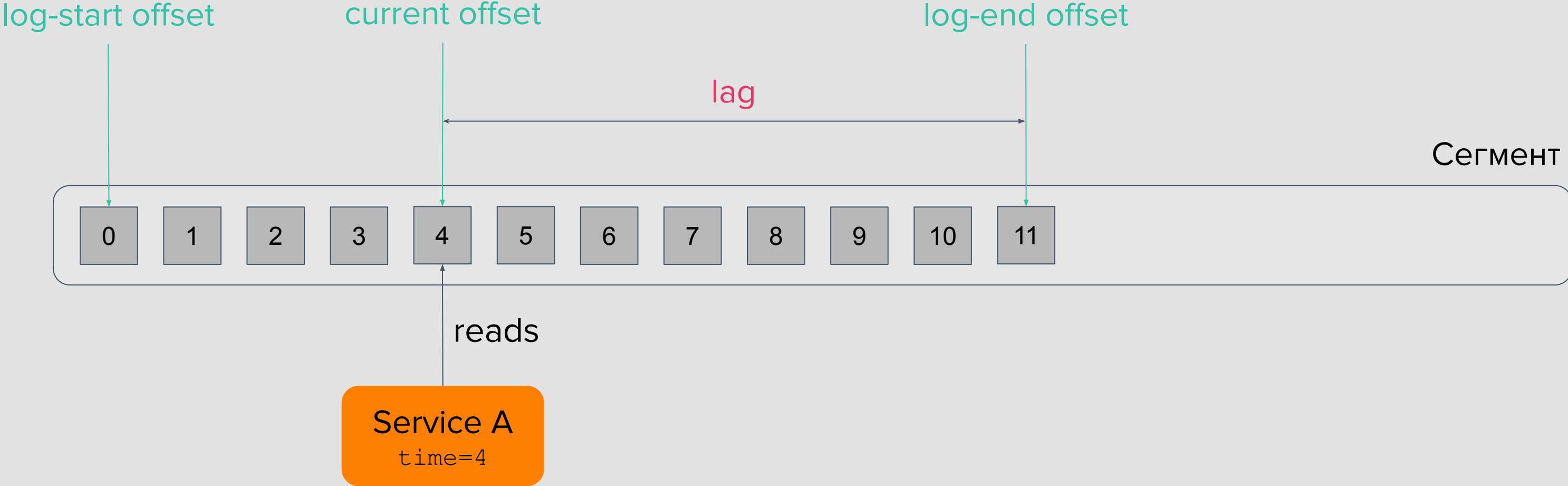
# Поток данных



# Поток данных

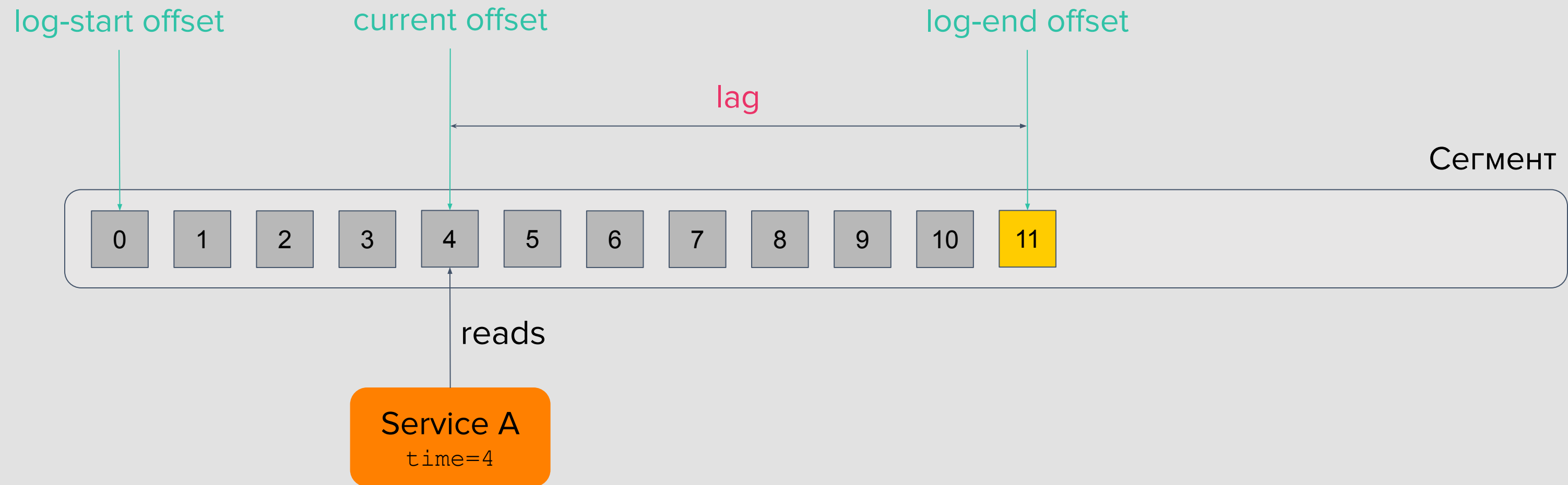


# Поток данных

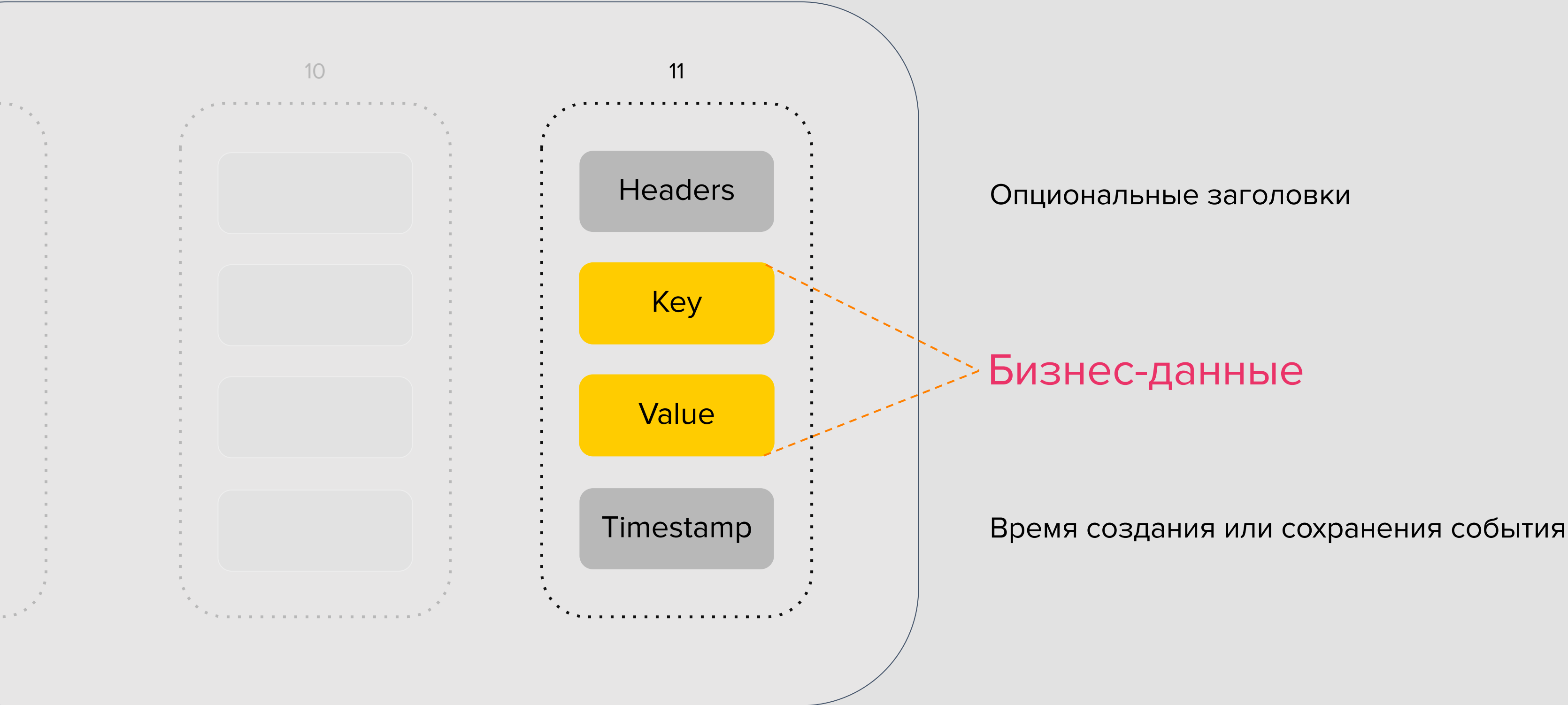




# Поток данных



# Сообщение в Kafka

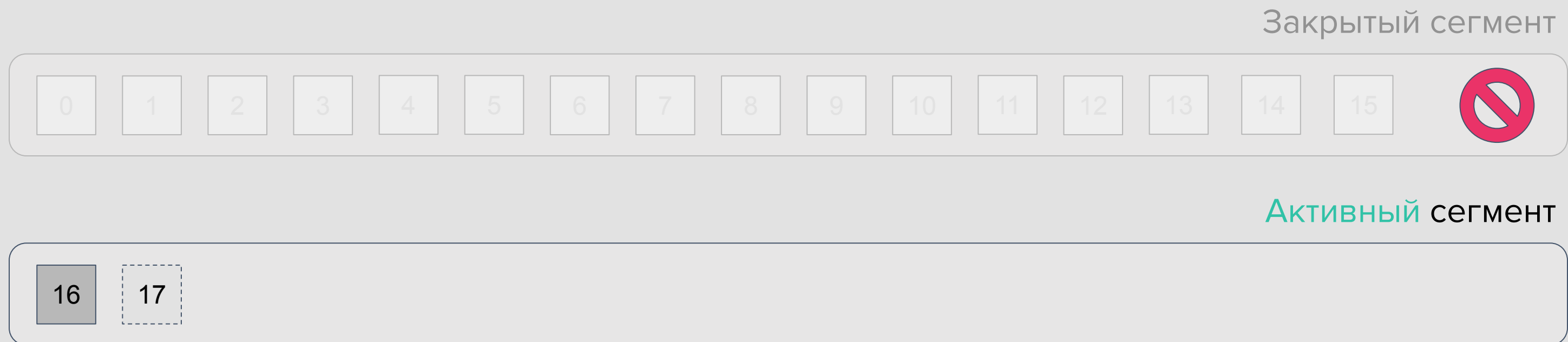


# Поток данных

Сегмент

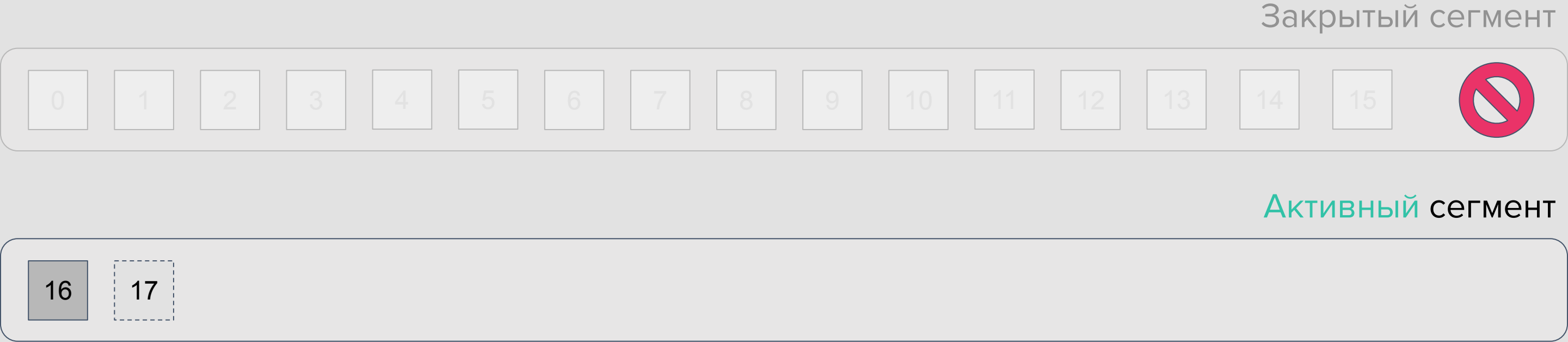


# Поток данных



# Устаревание данных

Закрытые сегменты удаляются по **времени, размеру или ключу**



# Устаревание данных

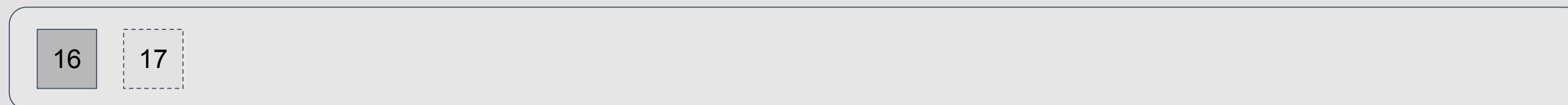
Закрытые сегменты удаляются по **времени, размеру или ключу**

Устаревание настраивается **глобально** или **на топик**

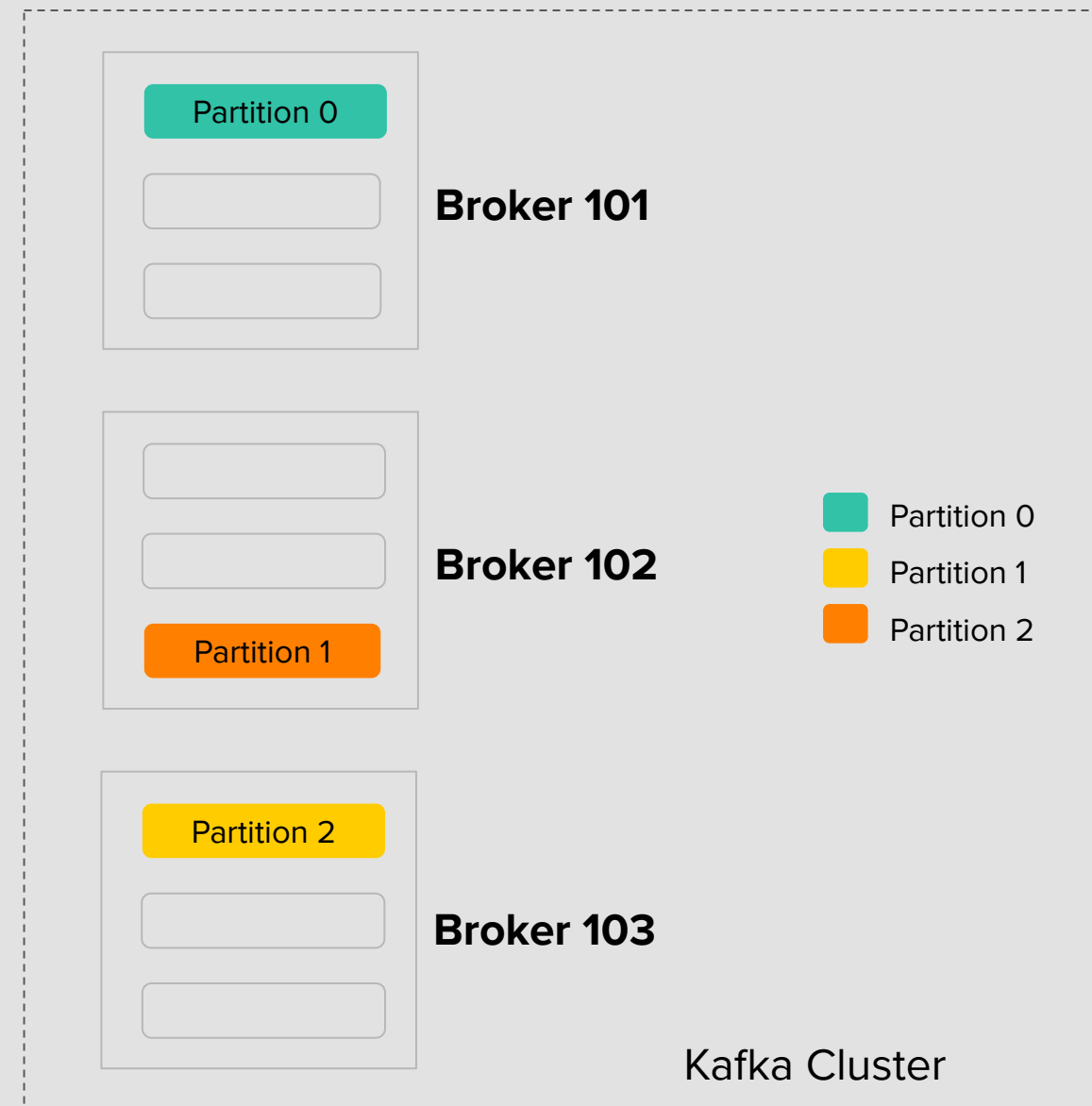
Закрытый сегмент



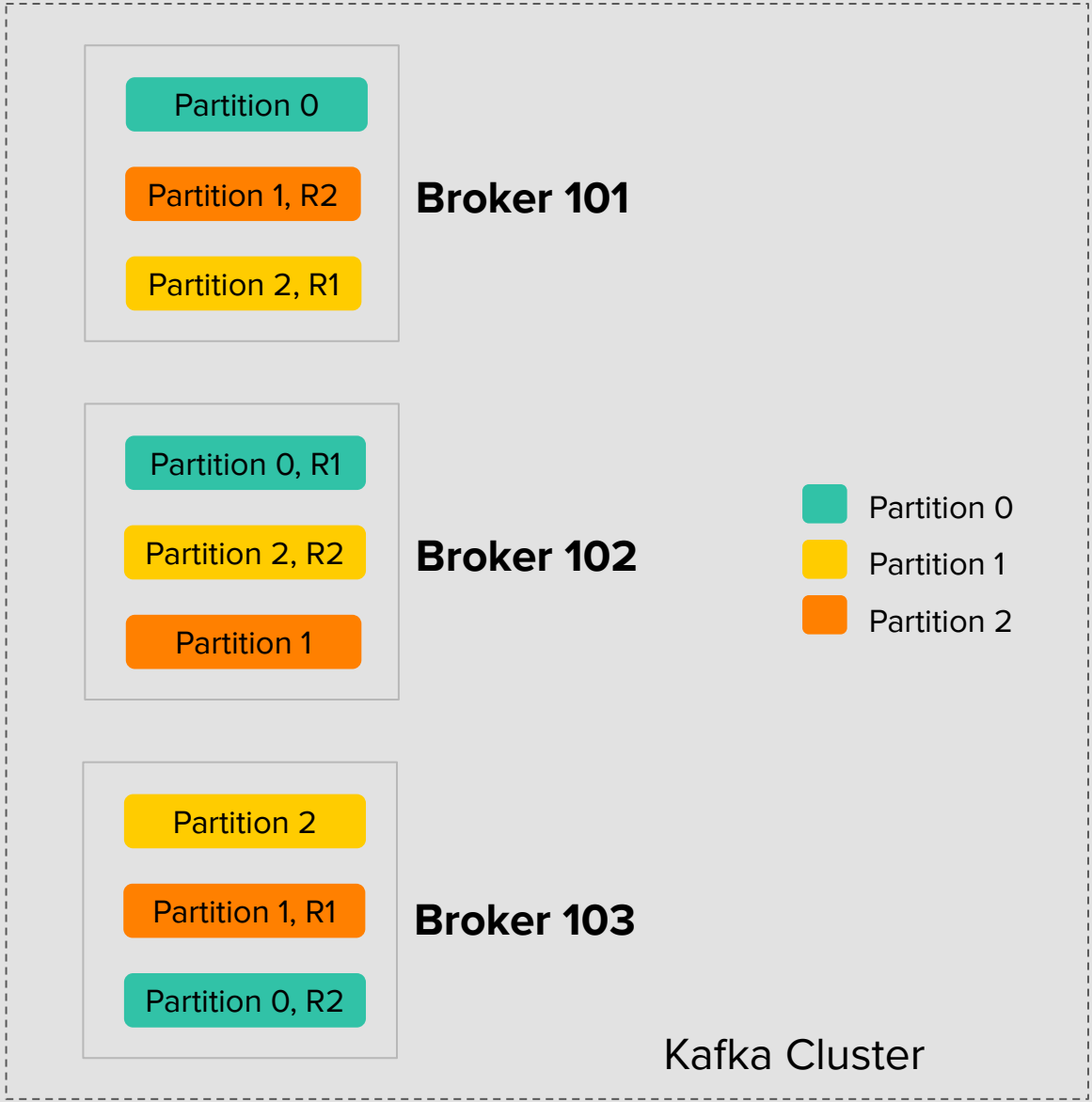
Активный сегмент



# Репликация данных



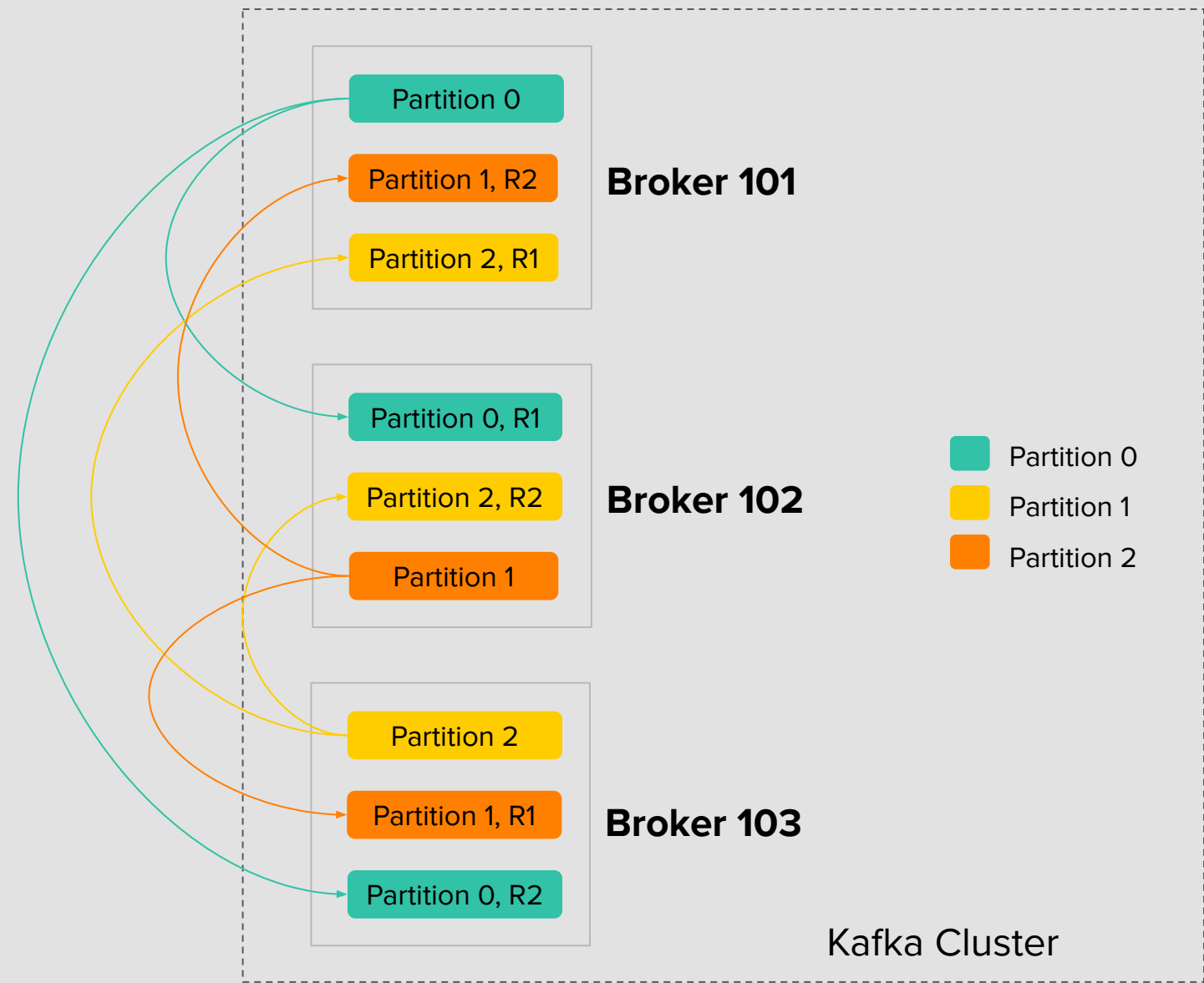
# Репликация данных



Репликация (с фактором 3)

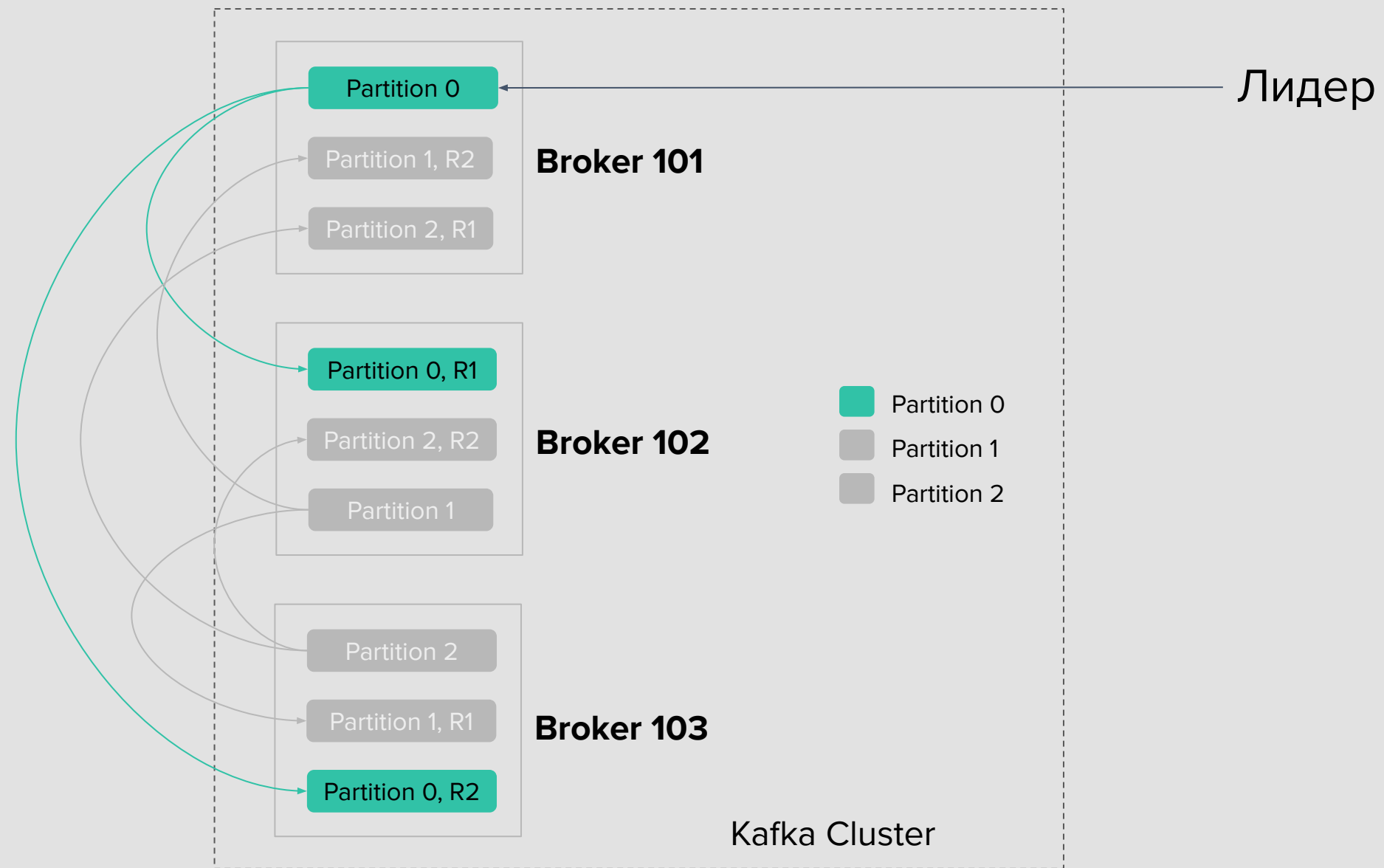


# Репликация данных



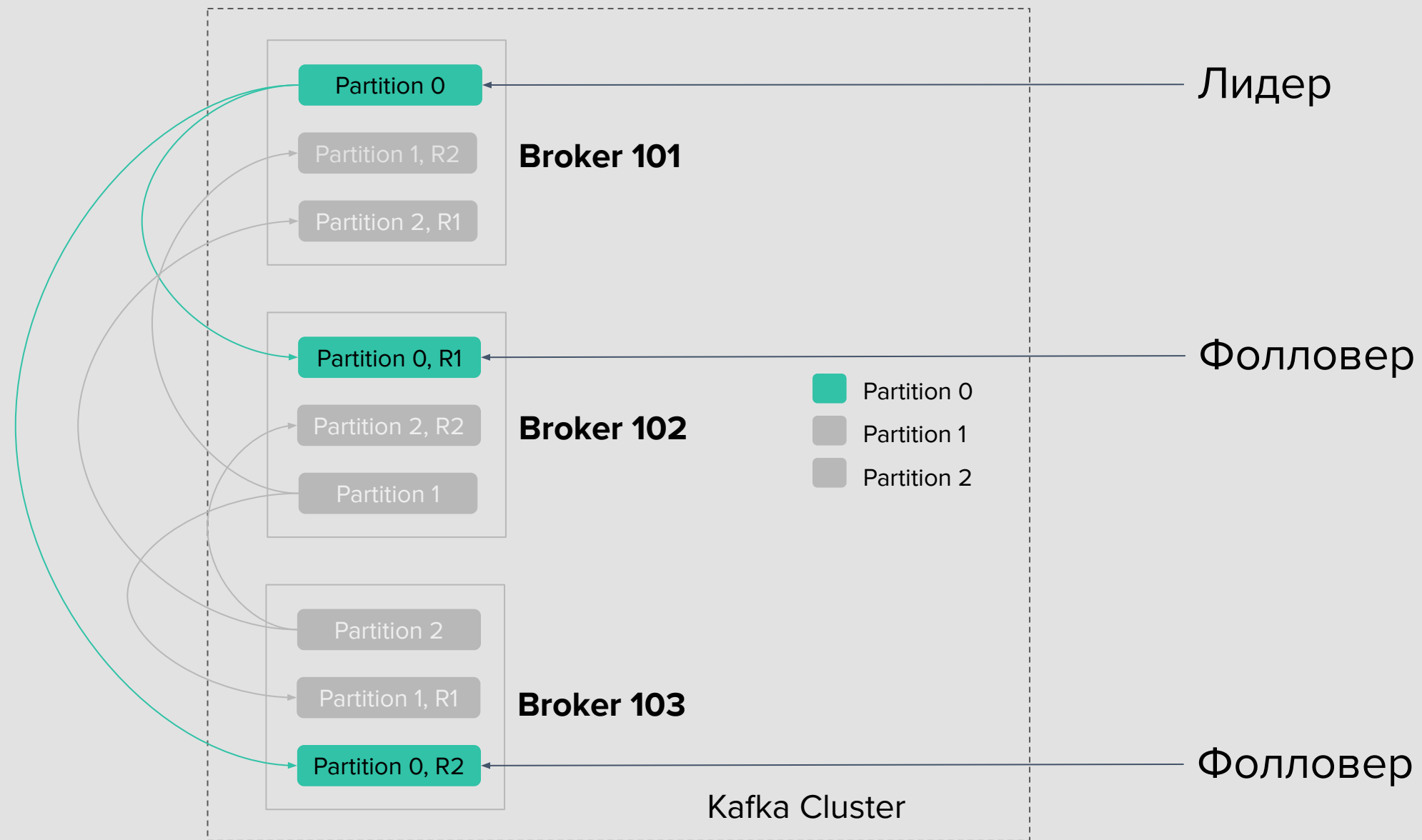
Репликация (с фактором 3)

# Репликация данных



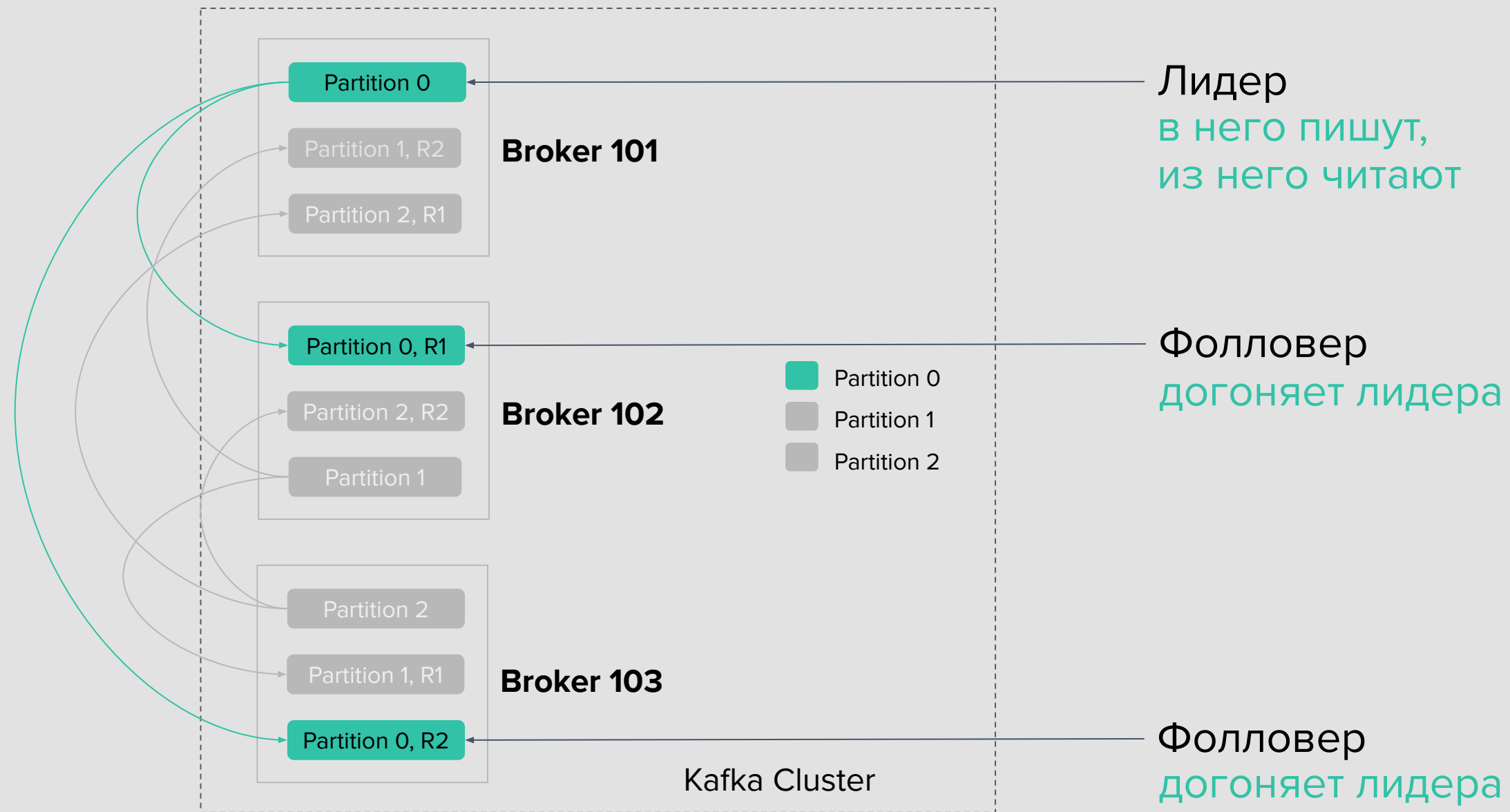
Репликация (с фактором 3)

# Репликация данных



Репликация (с фактором 3)

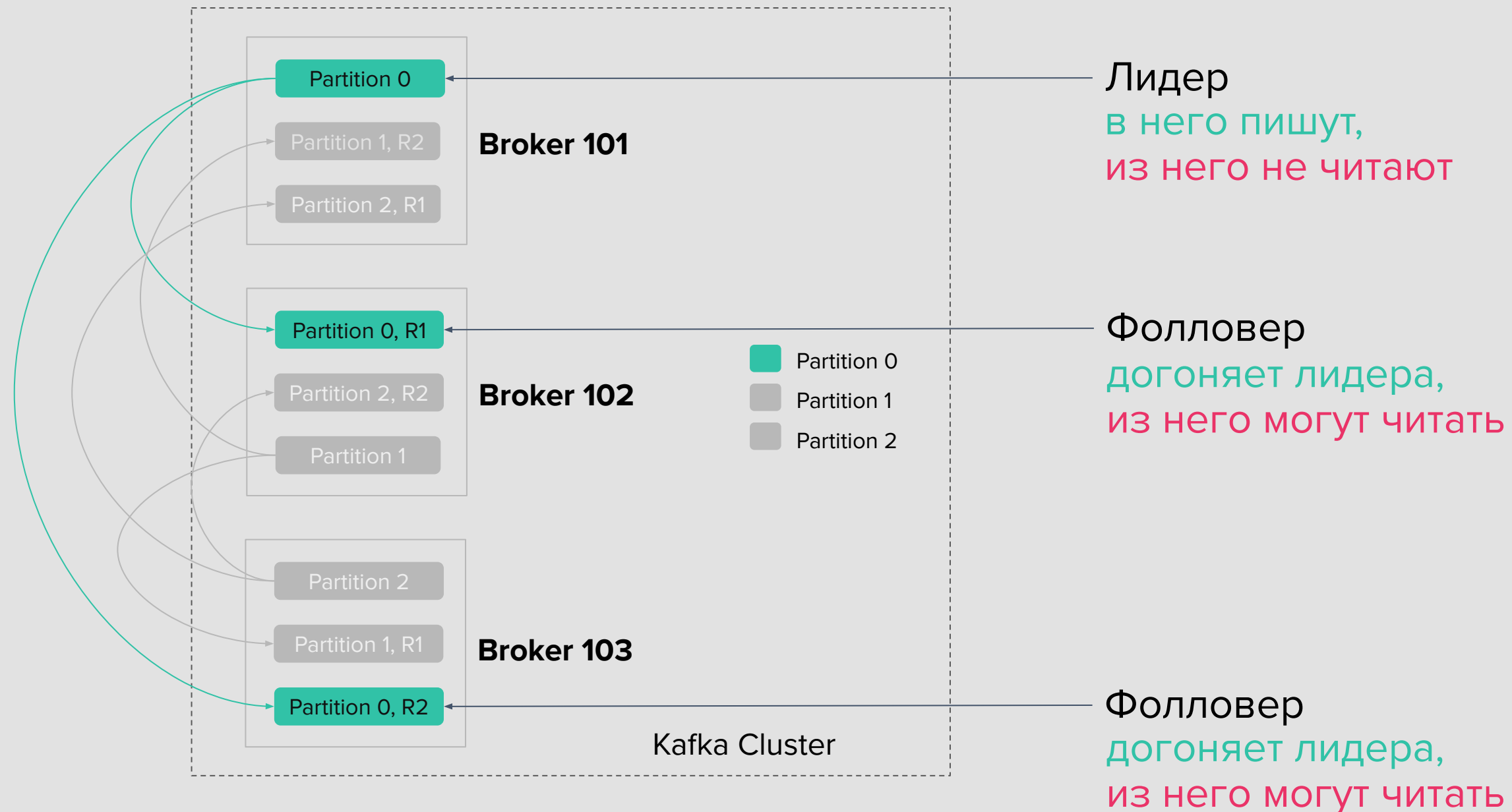
# Репликация данных



Репликация (с фактором 3)

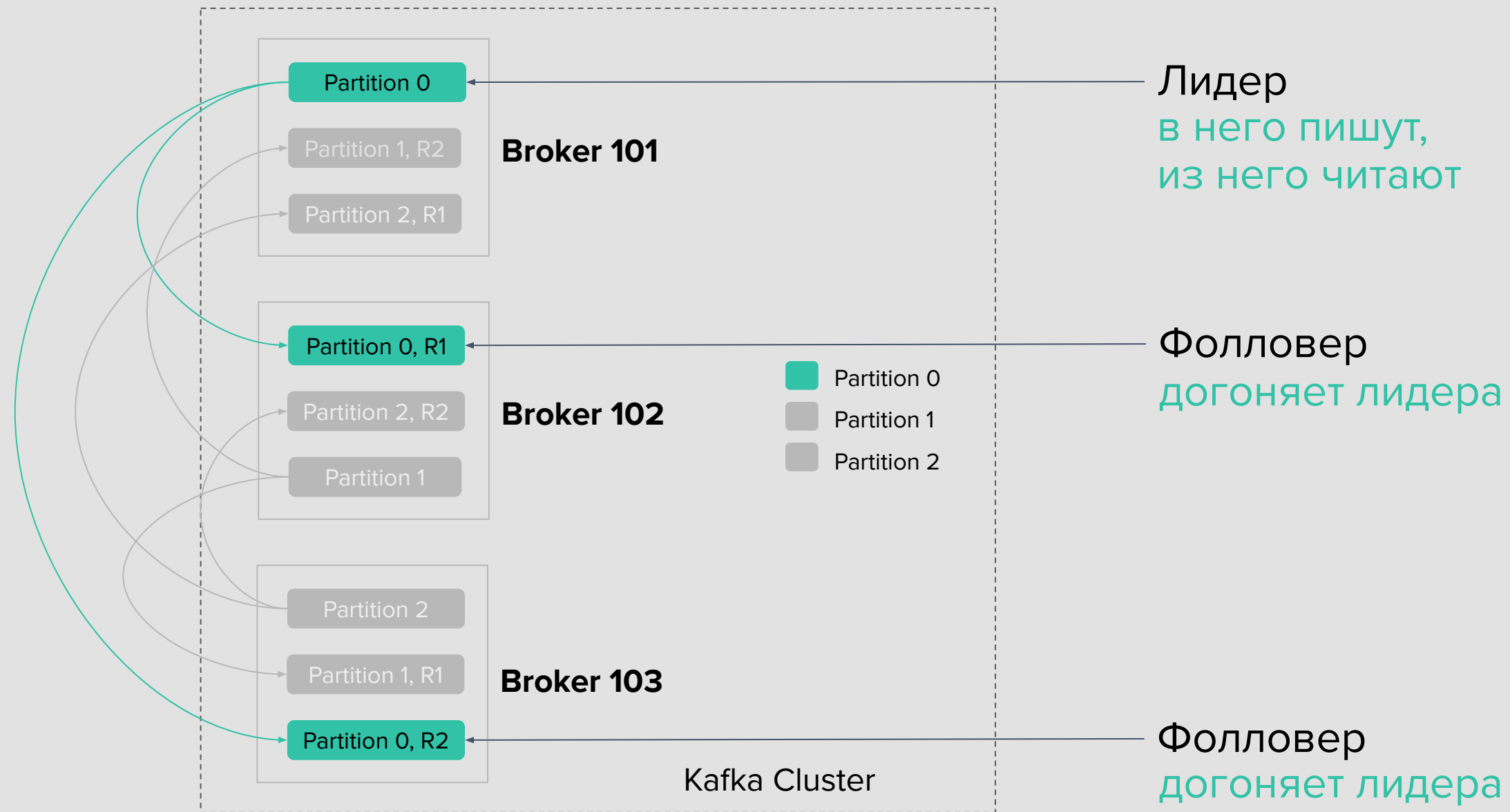
# Репликация данных

KIP-392: Allow consumers to fetch from closest replica



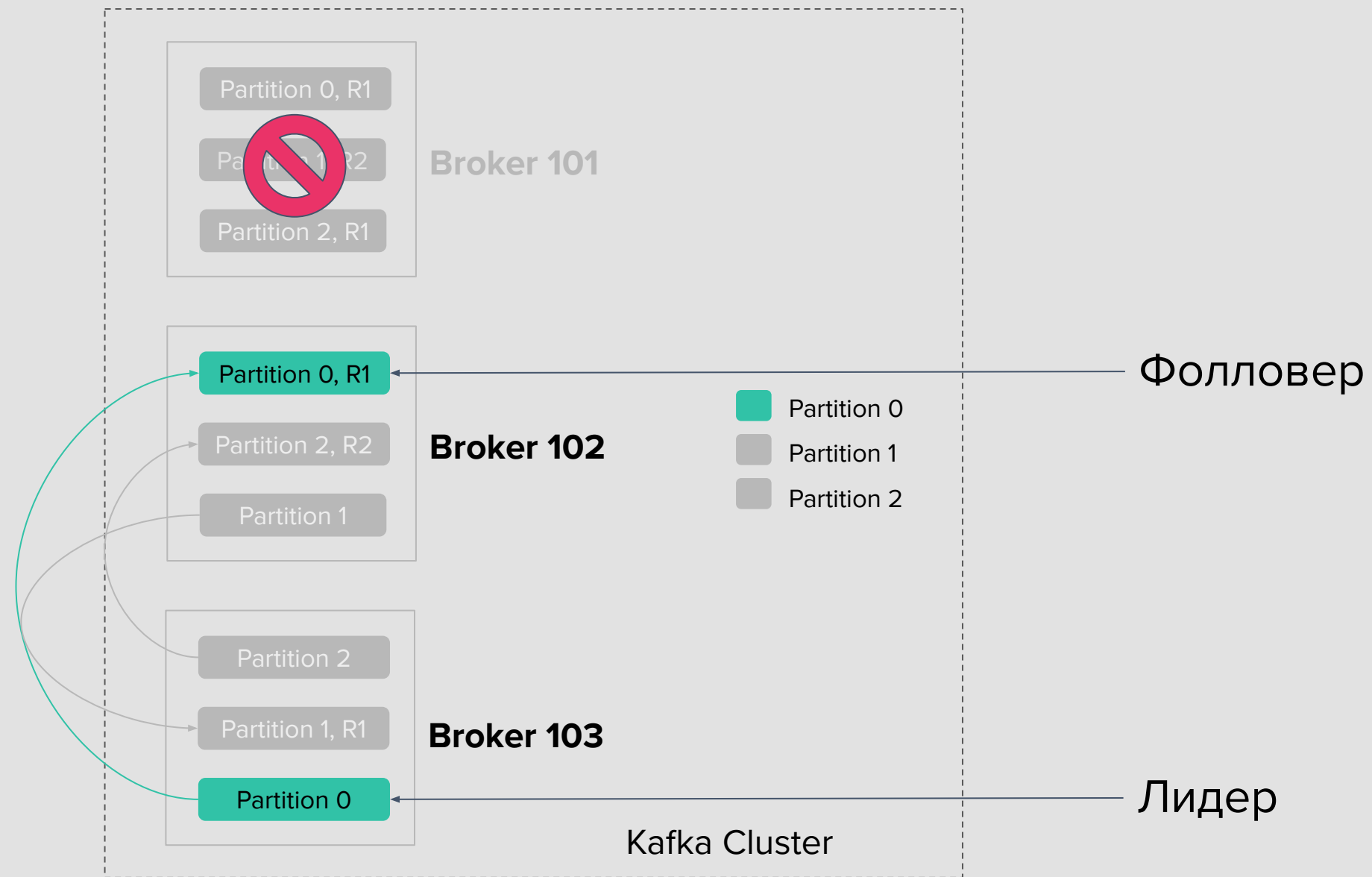
Репликация (с фактором 3)

# Репликация данных



Репликация (с фактором 3)

# Репликация данных



Репликация (с фактором 3)

# 01

Топики разделены на  
партиции



# 01

Топики разделены на  
партиции

# 02

Партиции находятся на  
одном или нескольких  
брокерах

# 01

Топики разделены на  
партиции

# 02

Партиции находятся на  
одном или нескольких  
брокерах

# 03

Партиция хранится на  
локальном диске  
брокера

# 01

Топики разделены на  
партиции

# 02

Партиции находятся на  
одном или нескольких  
брокерах

# 03

Партиция хранится на  
локальном диске  
брокера

# 04

Партиции представлены  
набором лог-файлов —  
сегментов

# 01

Топики разделены на  
партиции

# 02

Партиции находятся на  
одном или нескольких  
брокерах

# 03

Партиция хранится на  
локальном диске  
брокера

# 04

Партиции представлены  
набором лог-файлов —  
сегментов

# 05

Каждое сообщение в  
логе идентифицируется  
оффсетом

# 01

Топики разделены на  
партиции

# 02

Партиции находятся на  
одном или нескольких  
брокерах

# 03

Партиция хранится на  
локальном диске  
брокера

# 04

Партиции представлены  
набором лог-файлов —  
сегментов

# 05

Каждое сообщение в  
логе идентифицируется  
оффсетом

# 06

Лог-файлы устаревают  
по времени или размеру

# 01

Топики разделены на  
партиции

# 02

Партиции находятся на  
одном или нескольких  
брокерах

# 03

Партиция хранится на  
локальном диске  
брокера

# 04

Партиции представлены  
набором лог-файлов —  
сегментов

# 05

Каждое сообщение в  
логе идентифицируется  
оффсетом

# 06

Лог-файлы устаревают  
по времени или размеру

# 07

Партиции могут  
реплицироваться  
между брокерами

# 01

Топики разделены на партиции

# 02

Партиции находятся на одном или нескольких брокерах

# 03

Партиция хранится на локальном диске брокера

# 04

Партиции представлены набором лог-файлов — сегментов

# 05

Каждое сообщение в логе идентифицируется оффсетом

# 06

Лог-файлы устаревают по времени или размеру

# 07

Партиции могут реплицироваться между брокерами

# 08

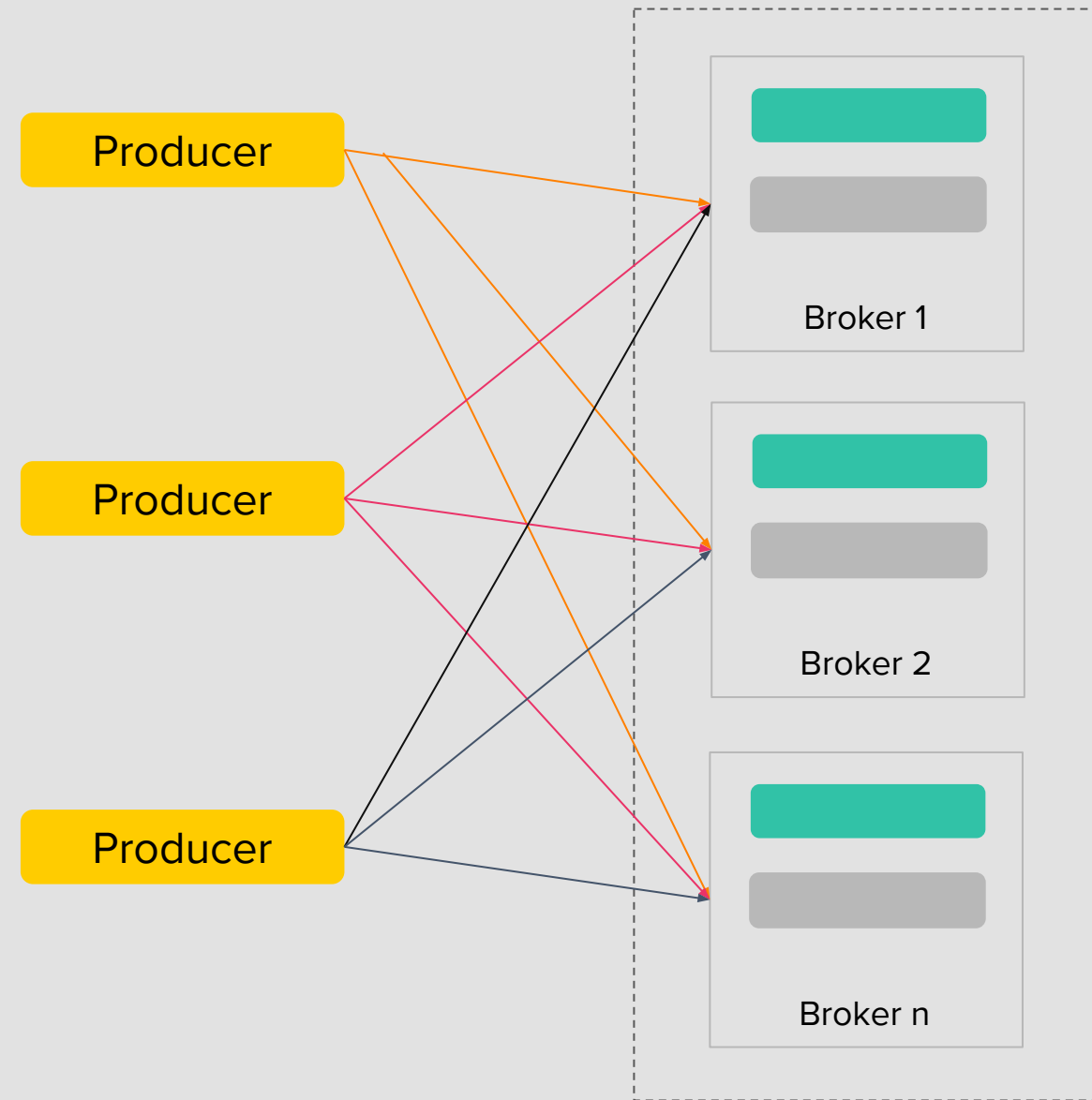
У партиции есть лидер и фолловеры. Консумеры и продюсеры работают с лидером.

# Продюсеры

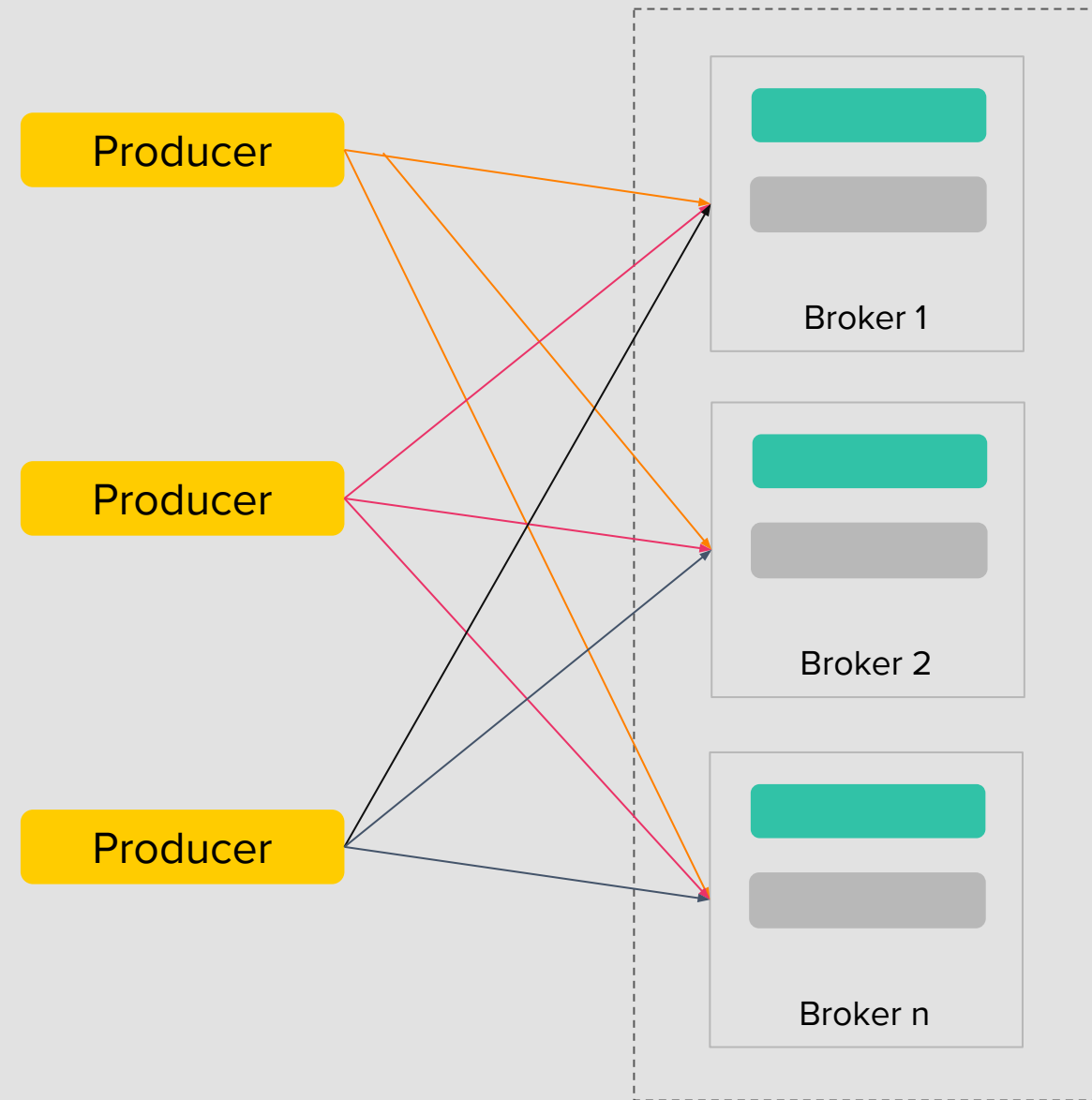




# Балансировка и партицирование

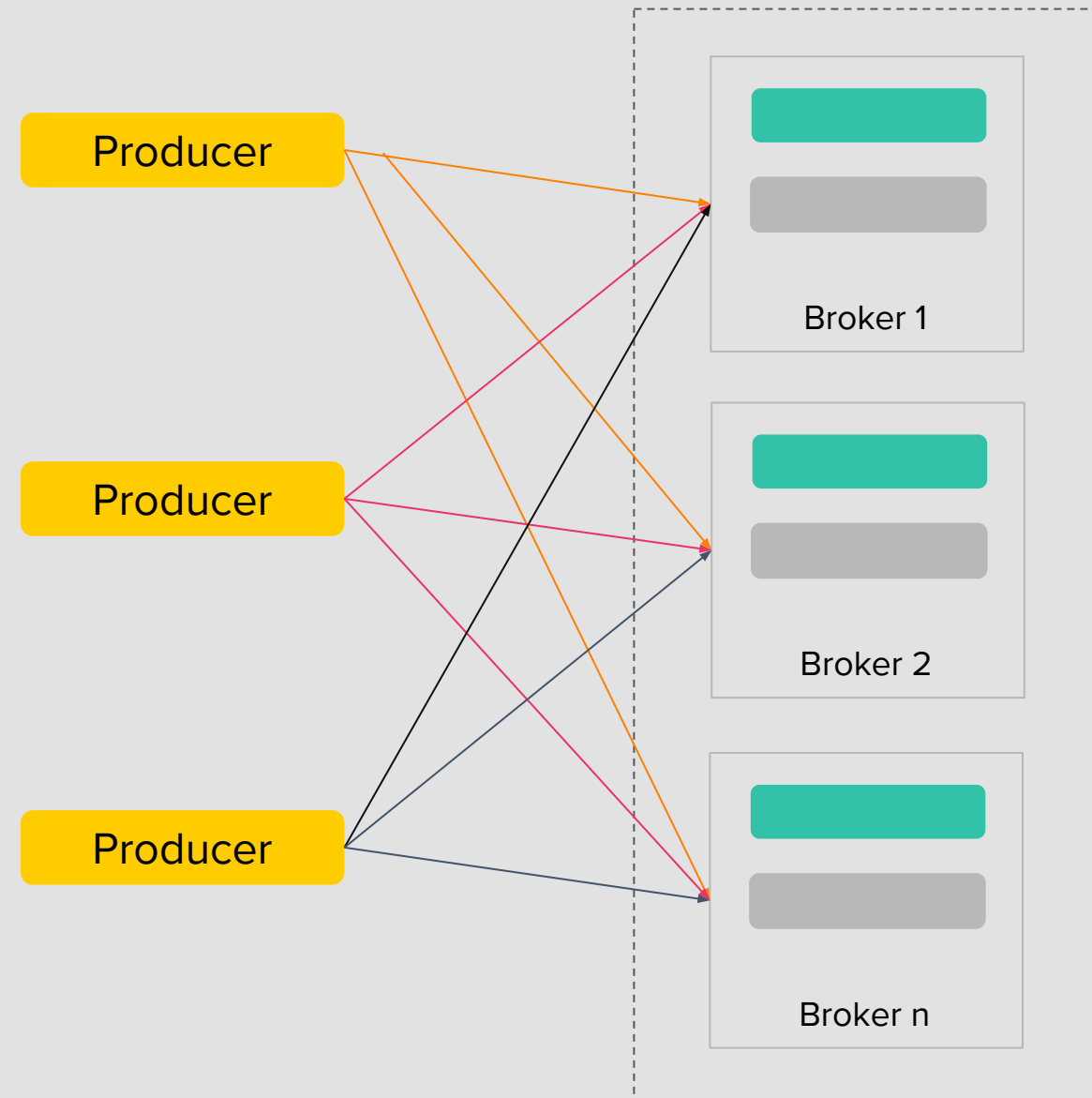


# Балансировка и партицирование



Продюсер определяет стратегию партицирования

# Балансировка и партицирование

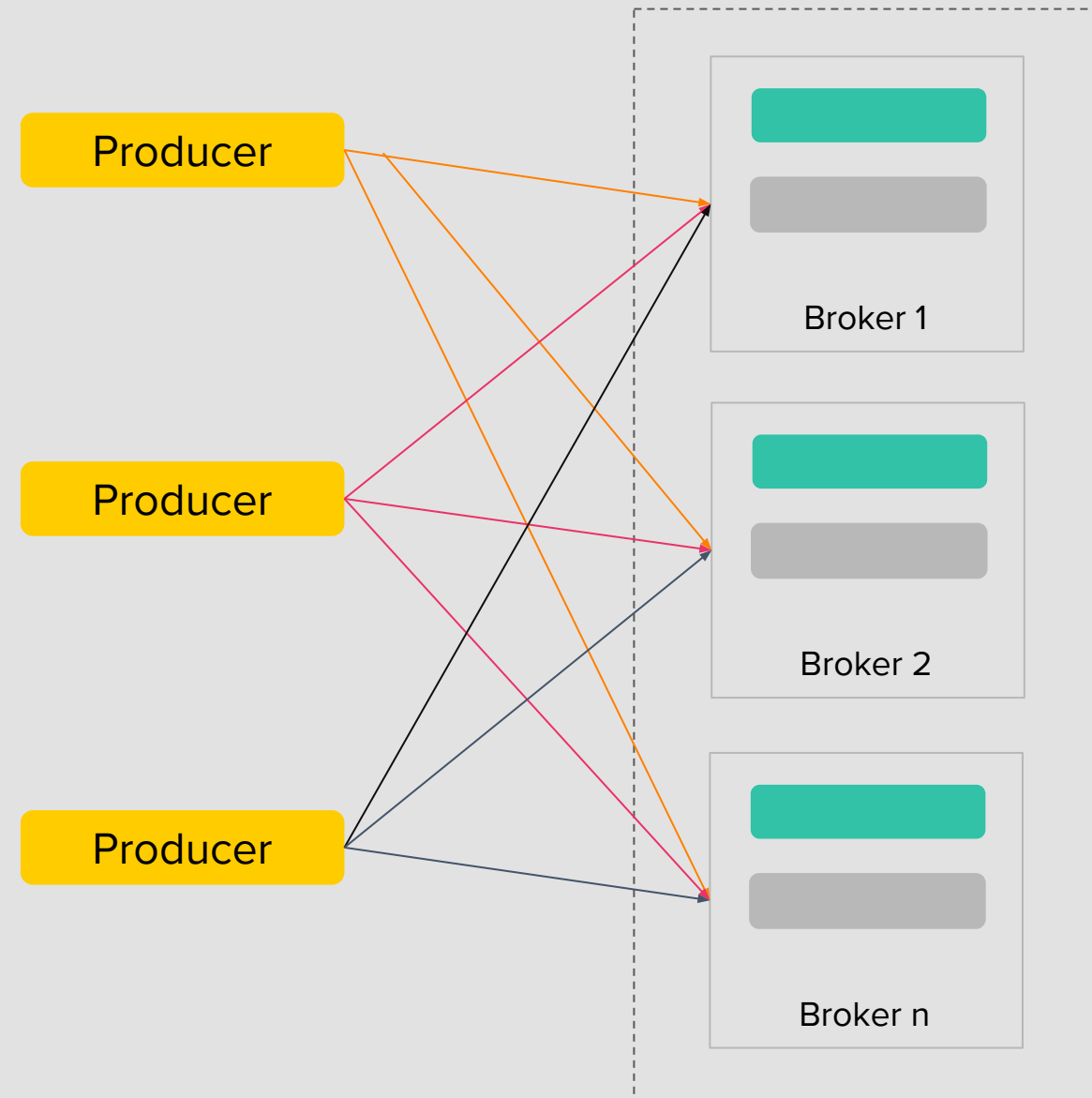


Продюсер определяет стратегию партицирования

**По ключу**

```
hash(key) % number_of_partitions
```

# Балансировка и партицирование



Продюсер определяет стратегию партицирования

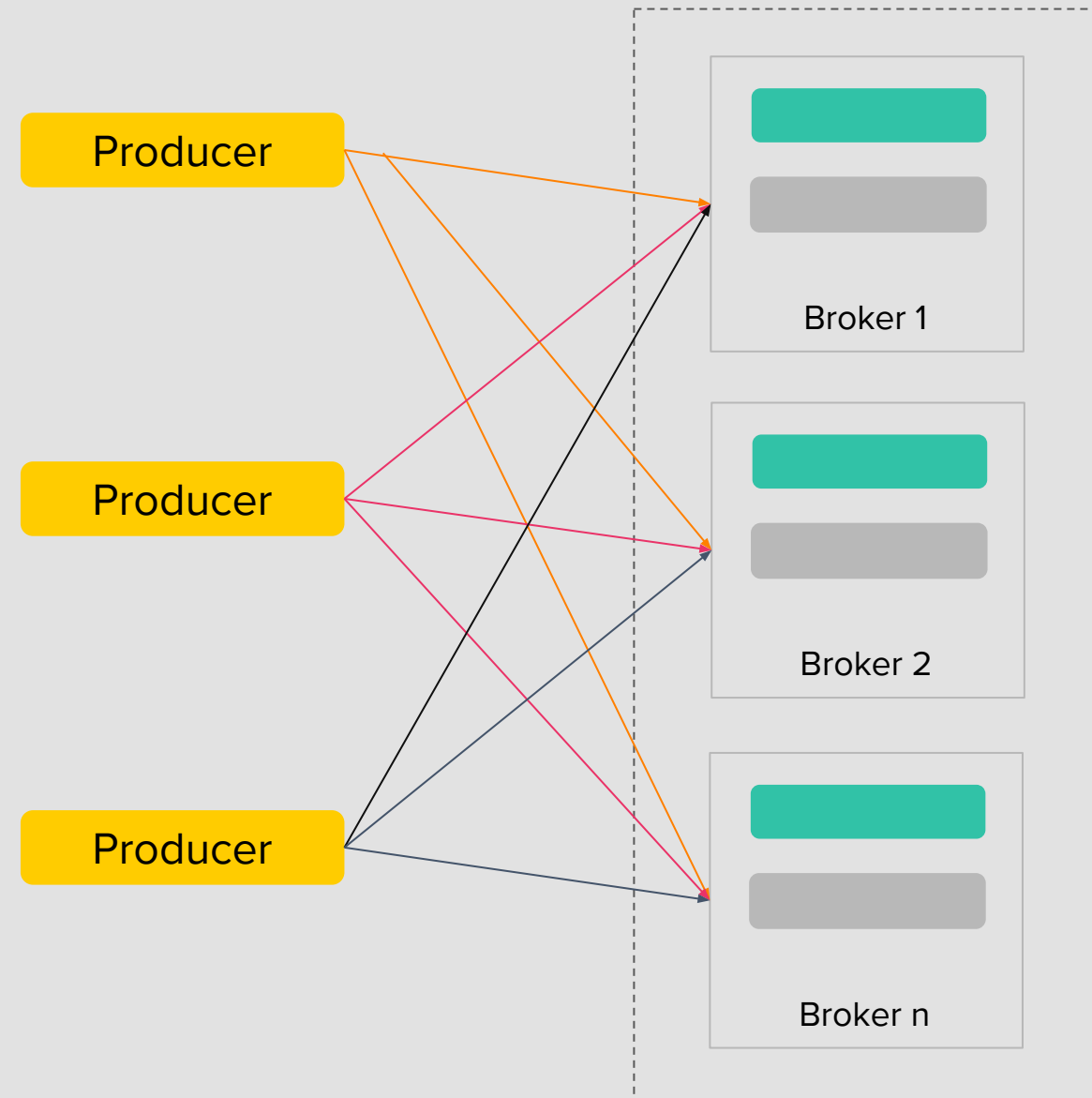
**По ключу**

`hash(key) % number_of_partitions`

**По очереди**

*без ключа партицирования*

# Балансировка и партицирование



Продюсер определяет стратегию партицирования

**По ключу**

`hash(key) % number_of_partitions`

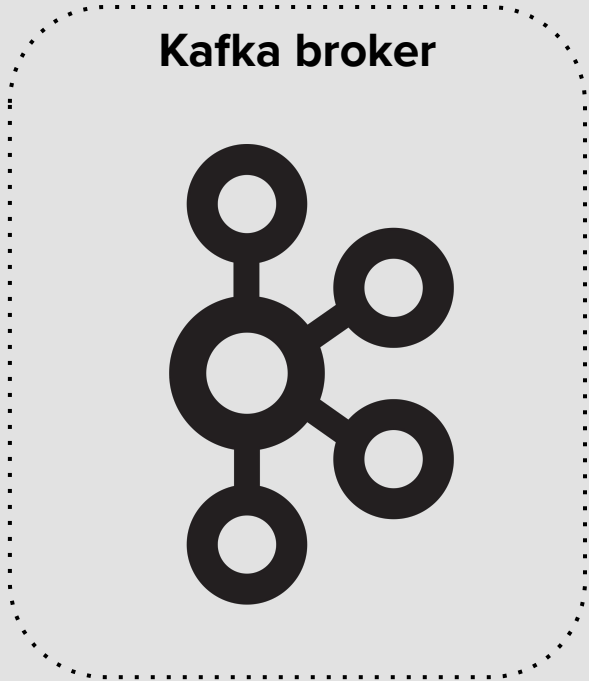
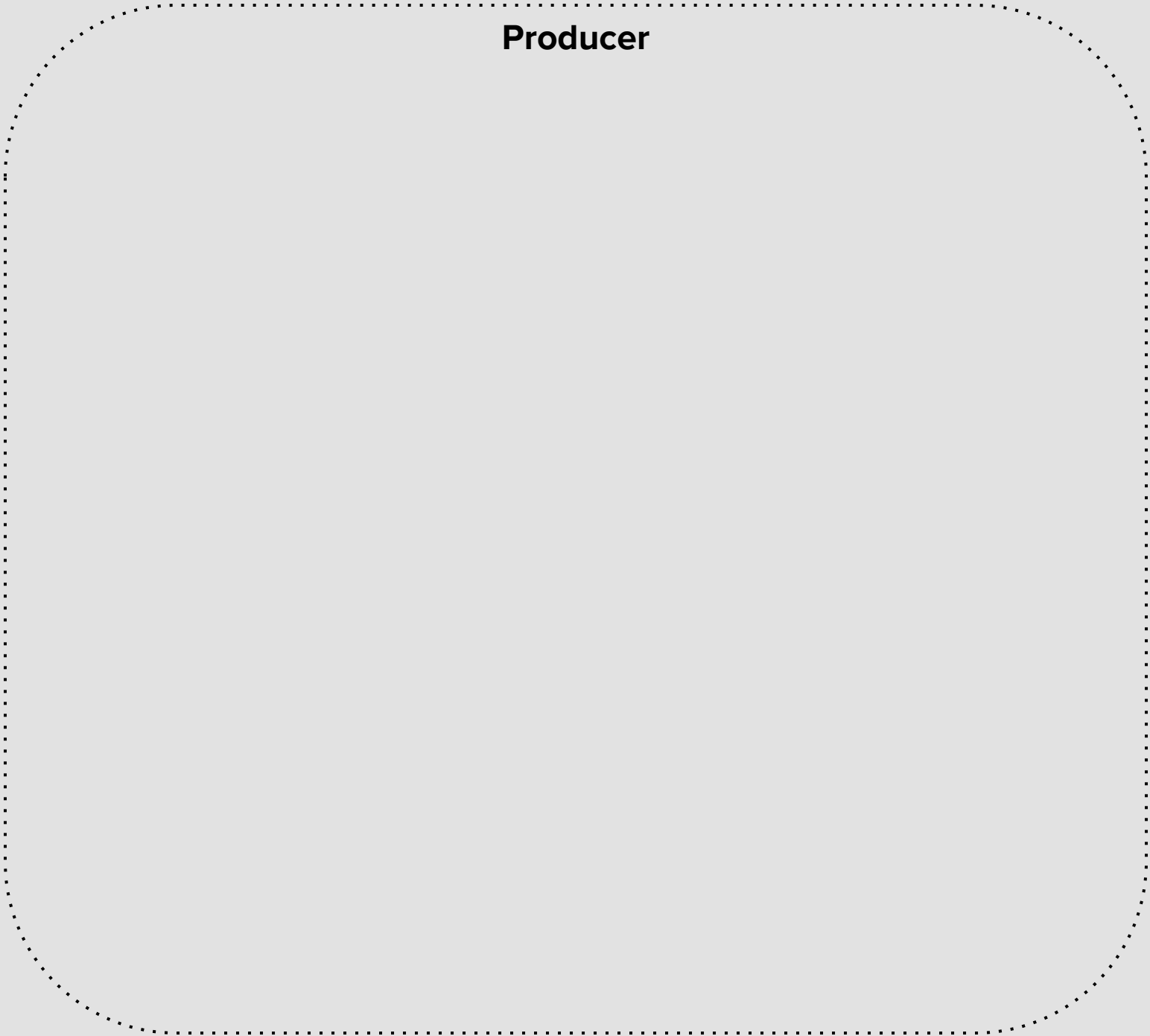
**По очереди**

*без ключа партицирования*

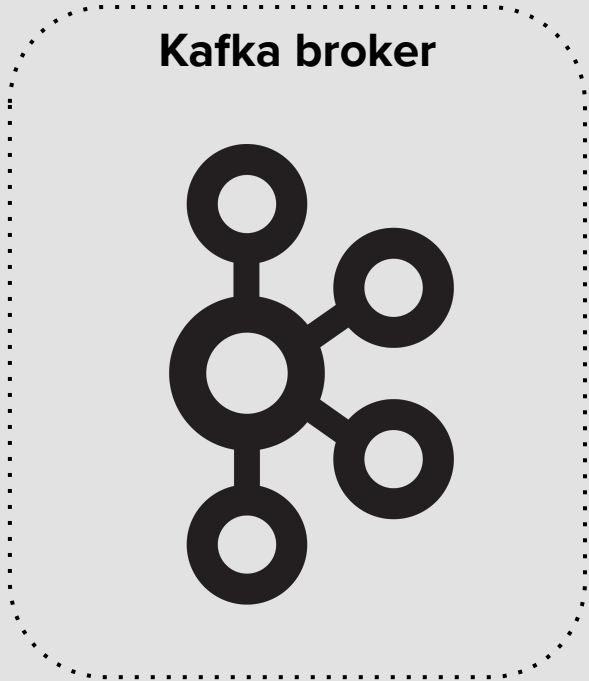
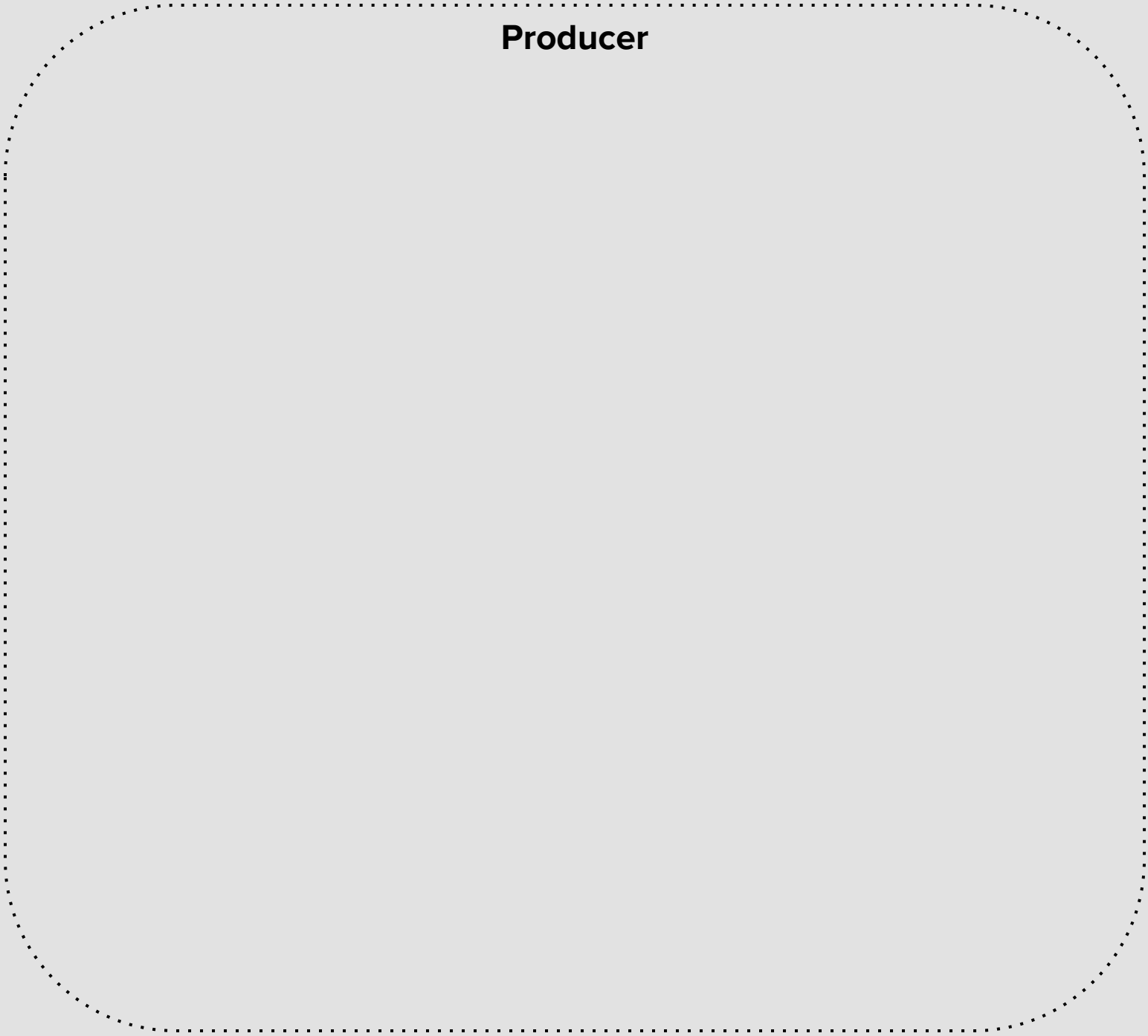
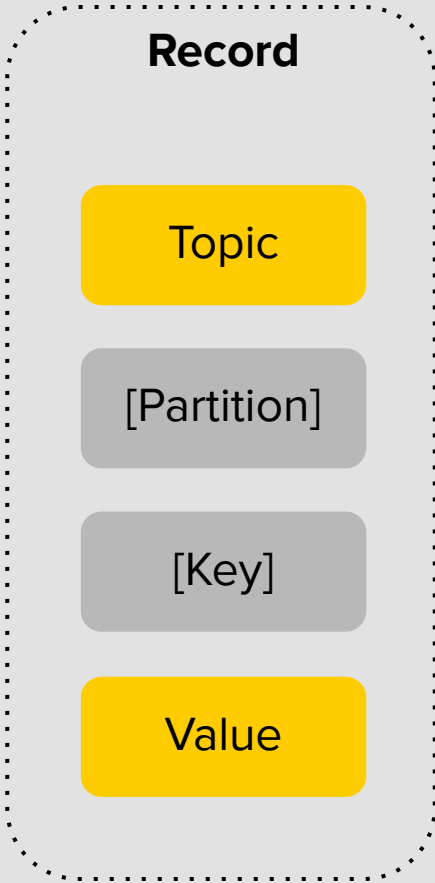
**Кастомная**

*реализуется продюсером*

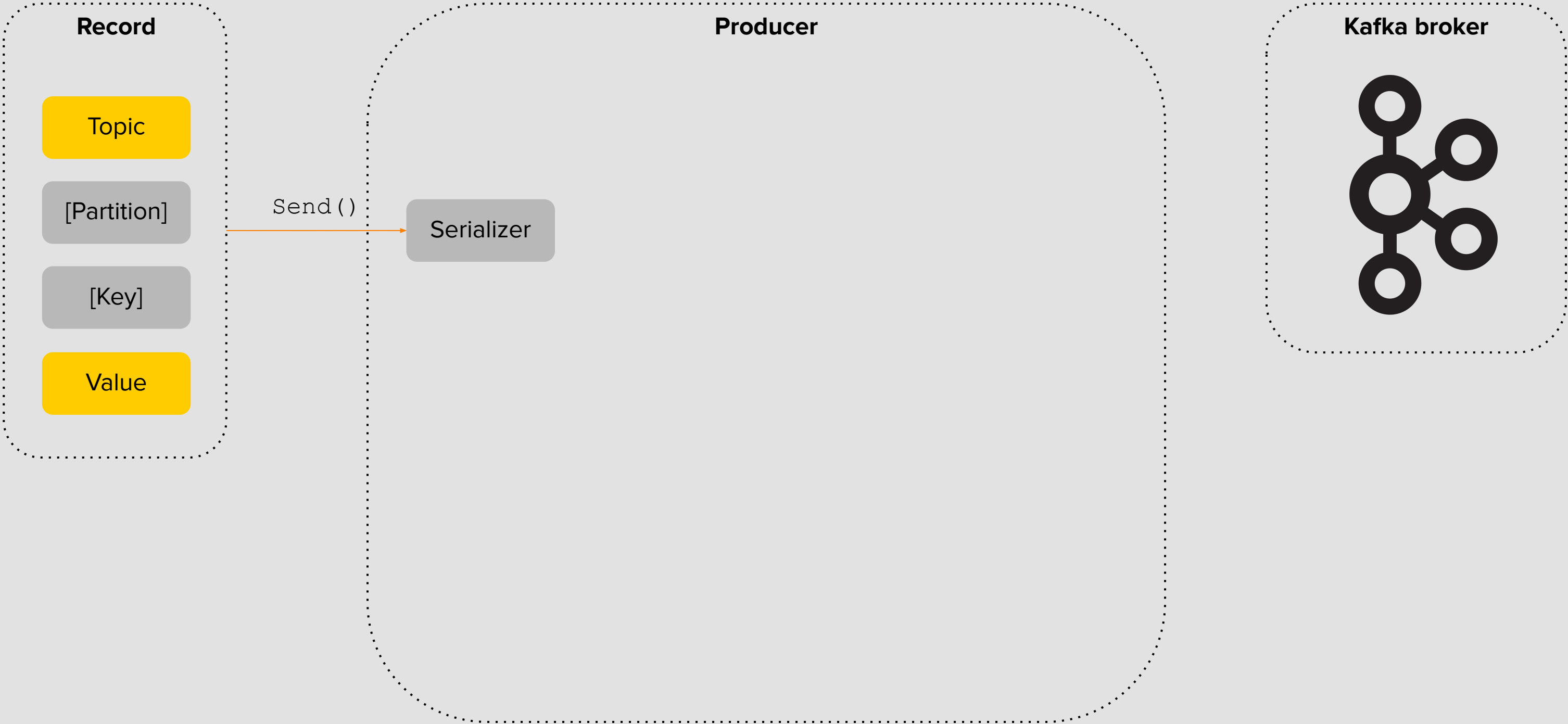
# Дизайн продюсера



# Дизайн продюсера

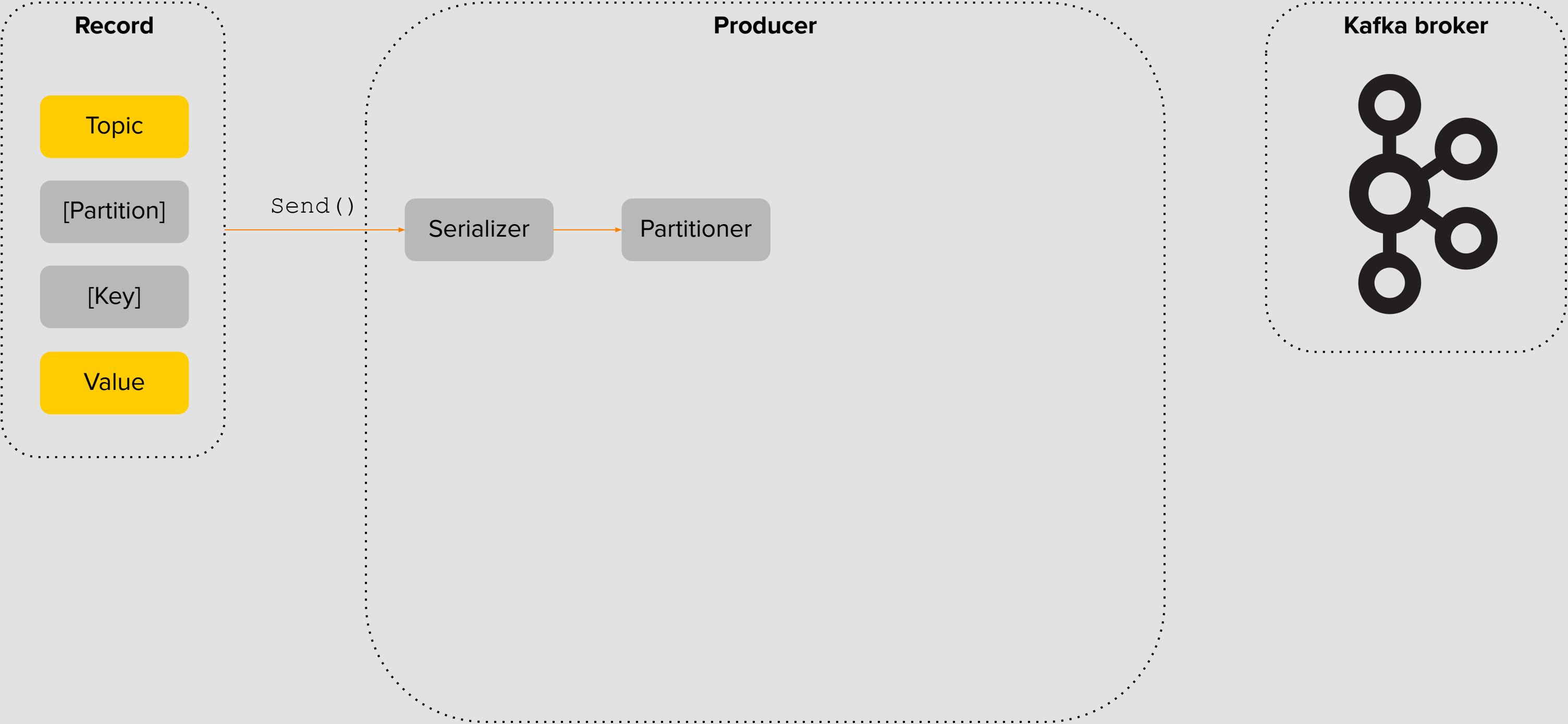


# Дизайн продюсера

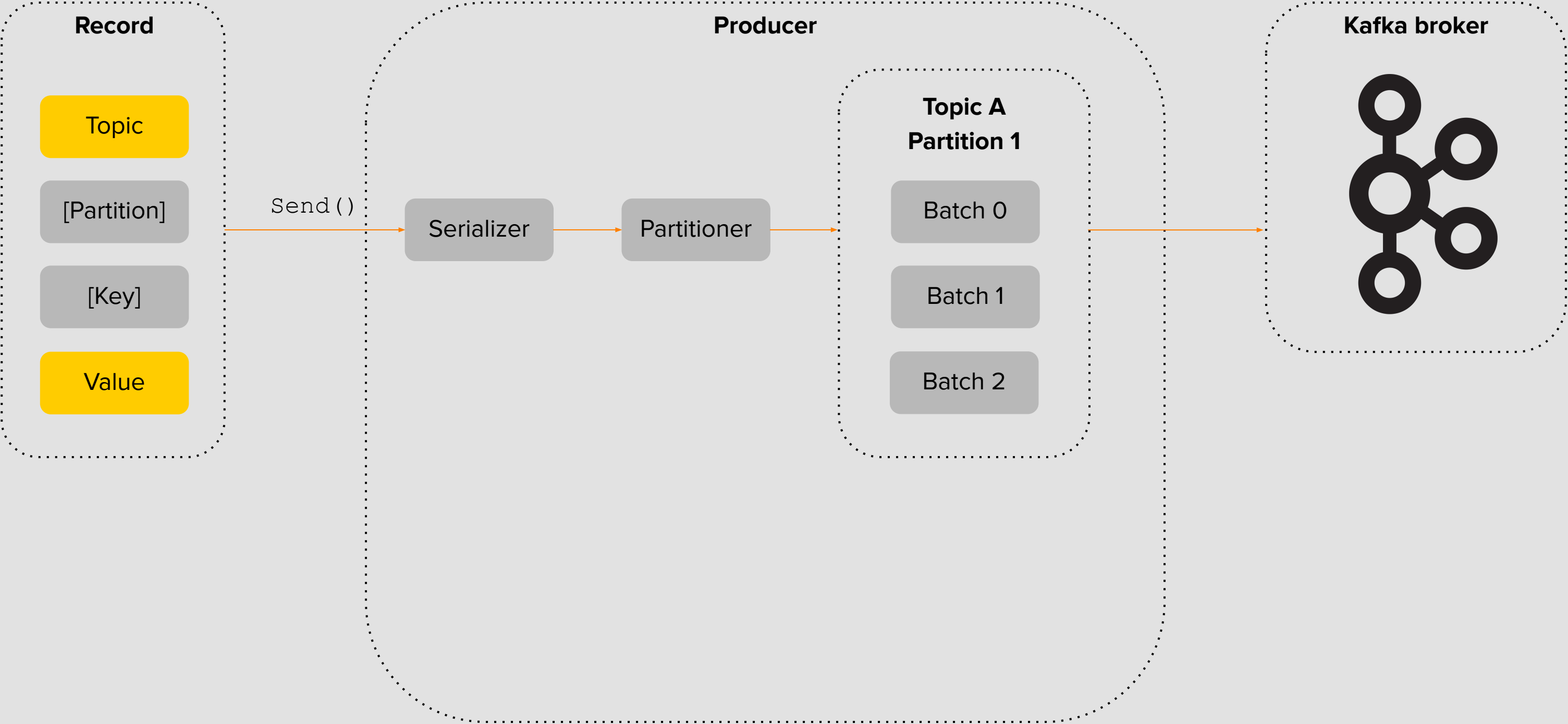




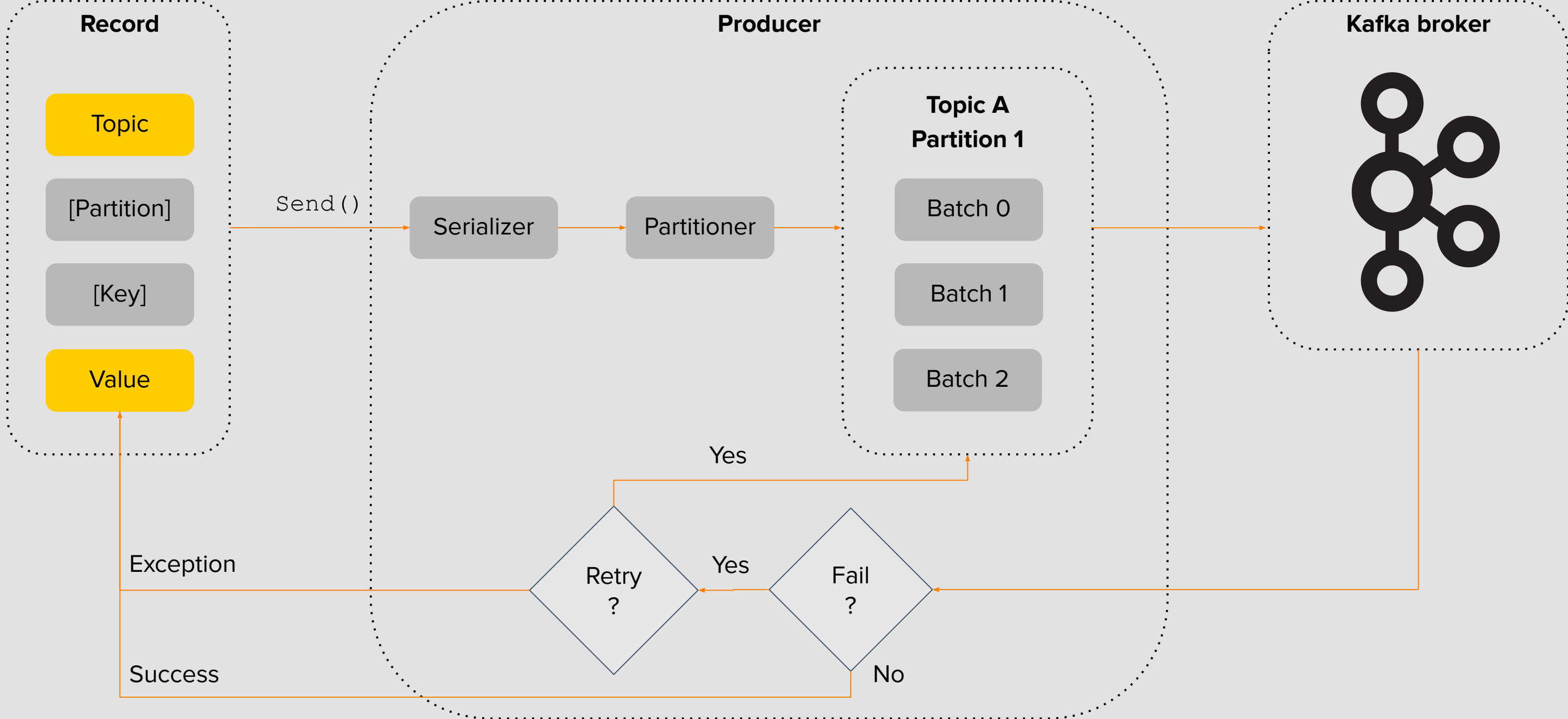
# Дизайн продюсера



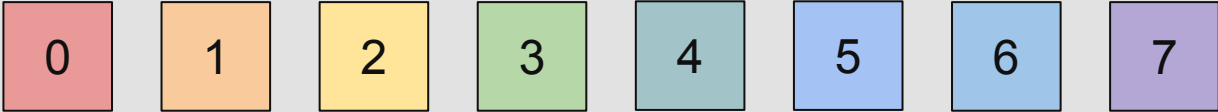
# Дизайн продюсера



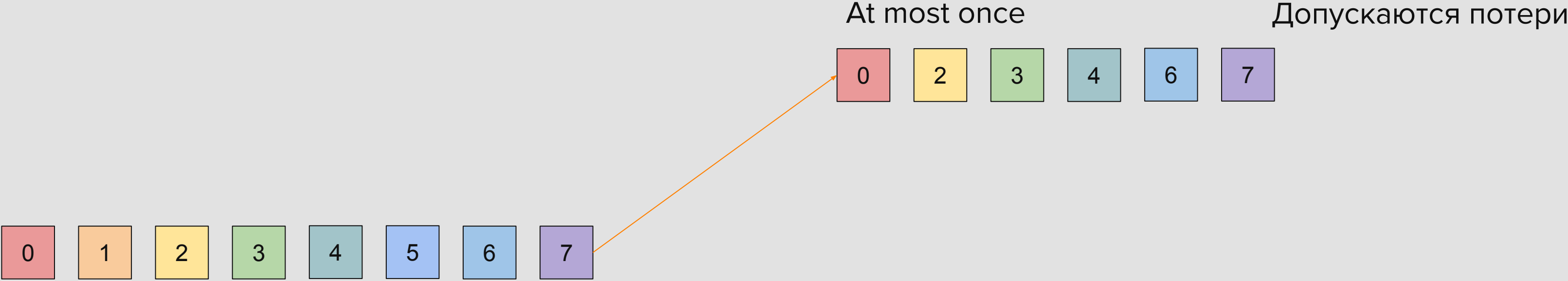
# Дизайн продюсера



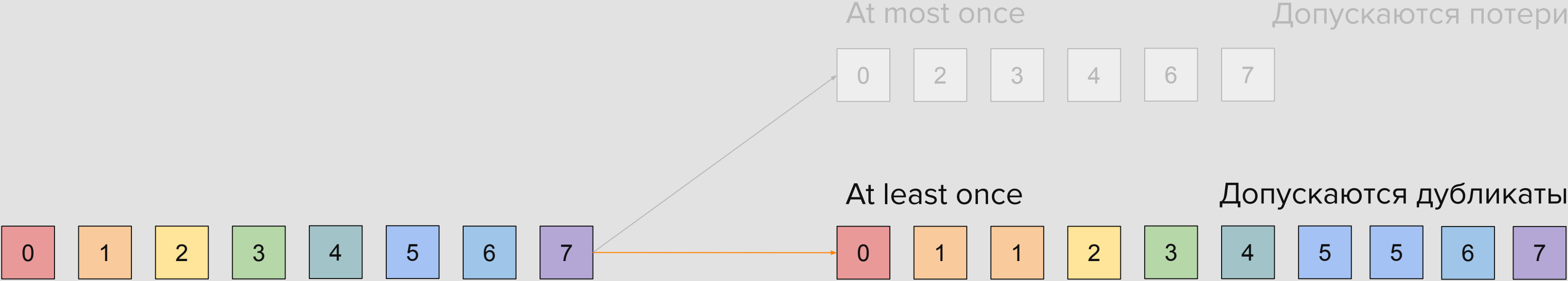
# Семантики доставки



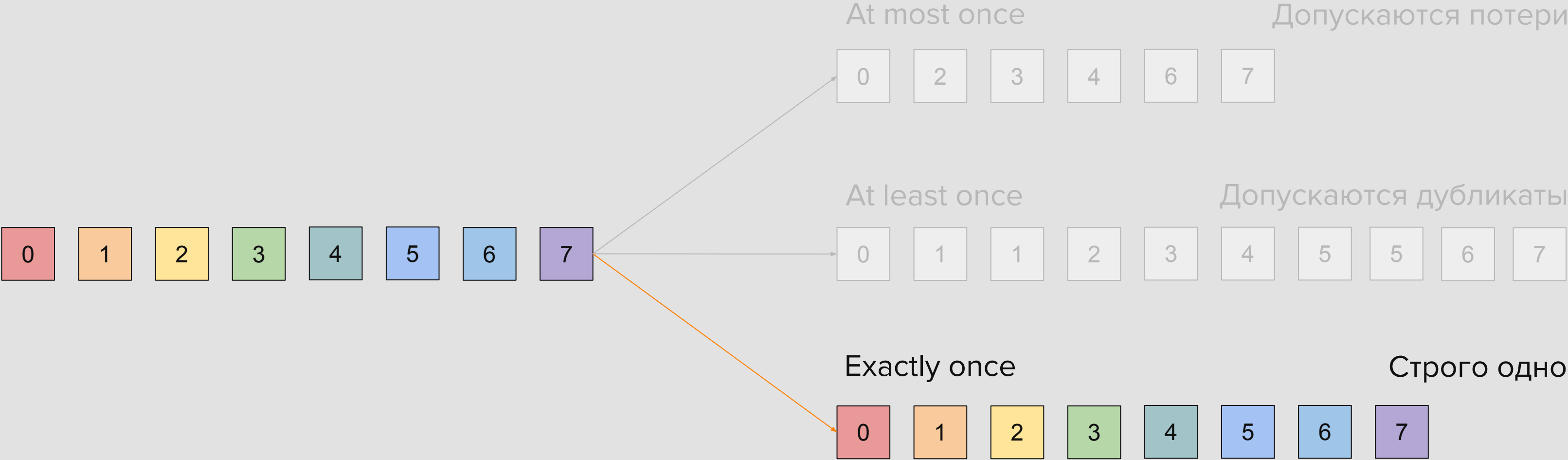
# Семантики доставки



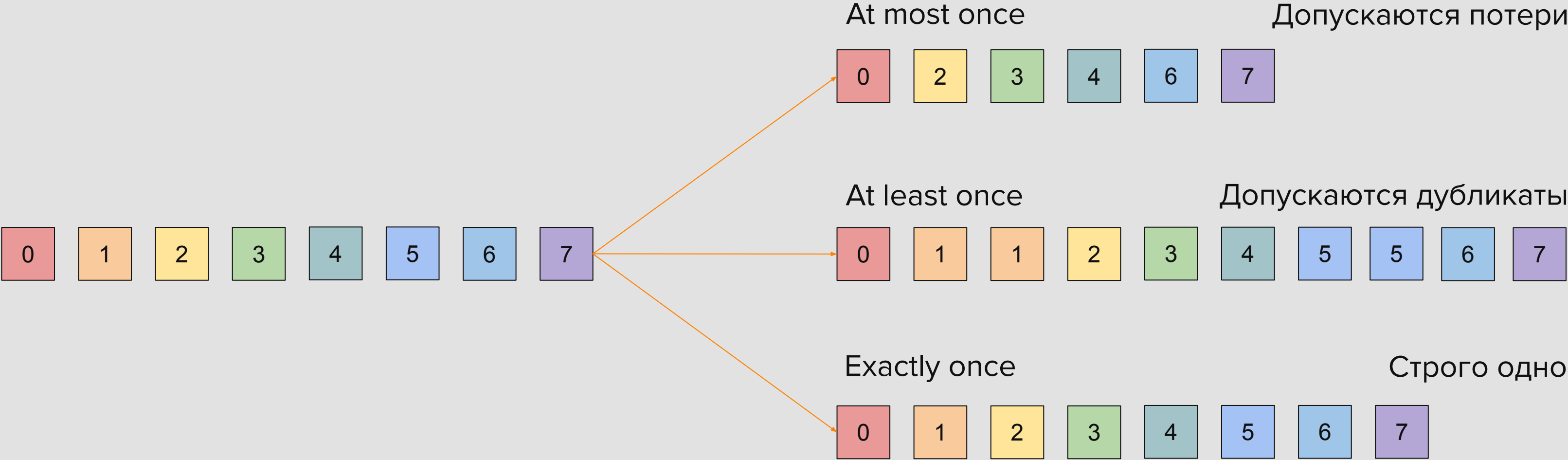
# Семантики доставки



# Семантики доставки

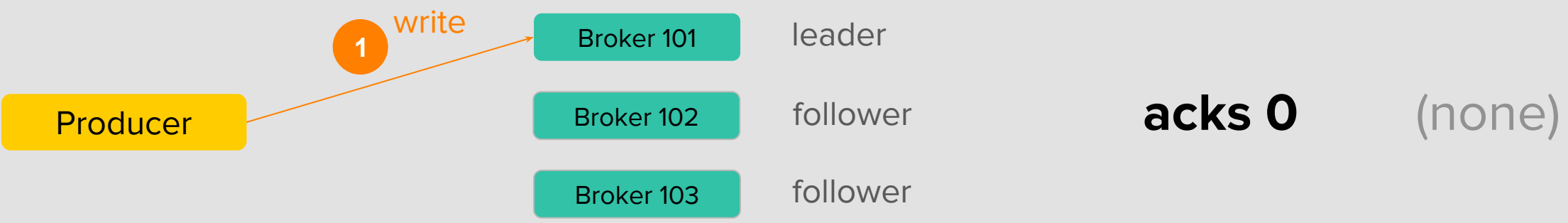


# Семантики доставки





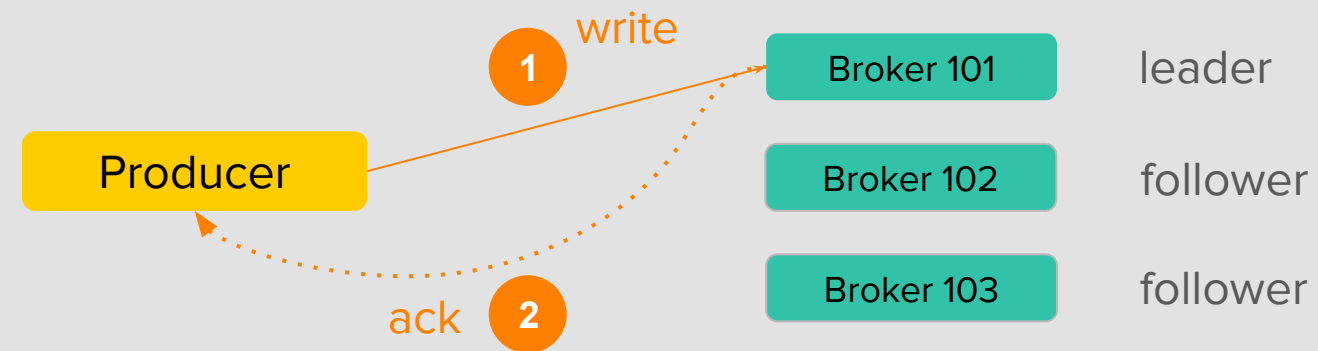
# Надёжность доставки



# Надёжность доставки

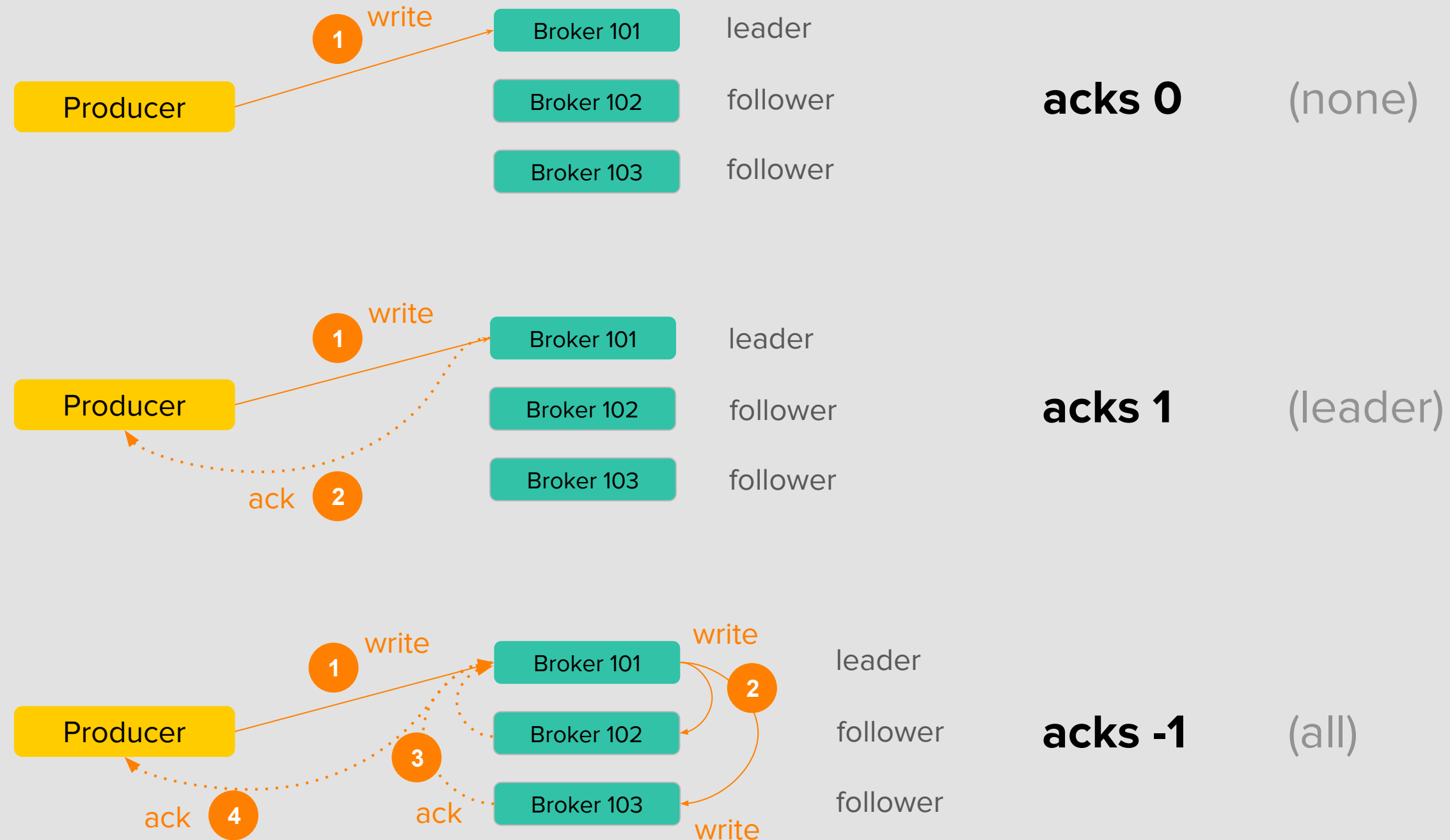


**acks 0** (none)



**acks 1** (leader)

# Надёжность доставки



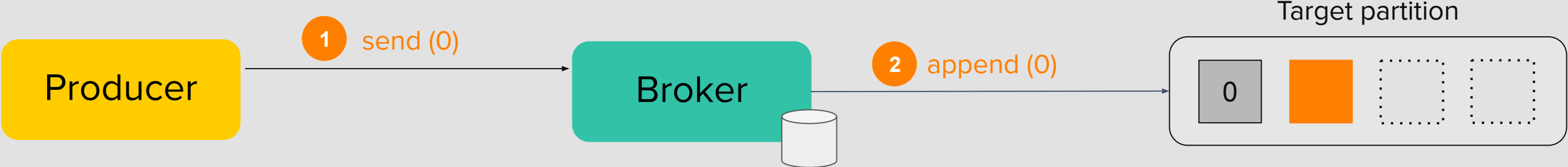
# Идемпотентные продюсеры

Штатная работа



# Идемпотентные продюсеры

## Штатная работа



# Идемпотентные продюсеры

## Штатная работа

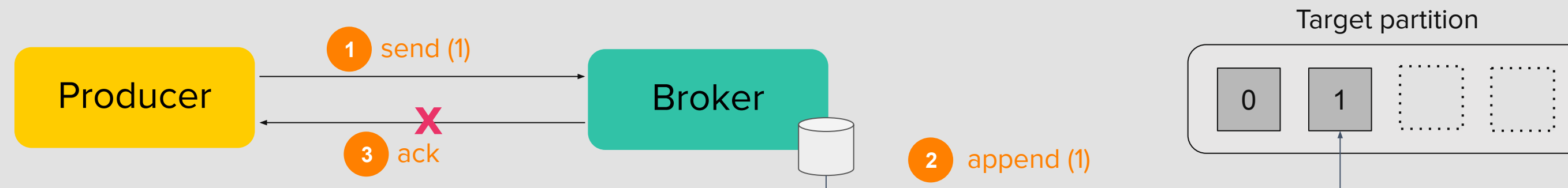


# Идемпотентные продюсеры

## Штатная работа



## Сбой при записи

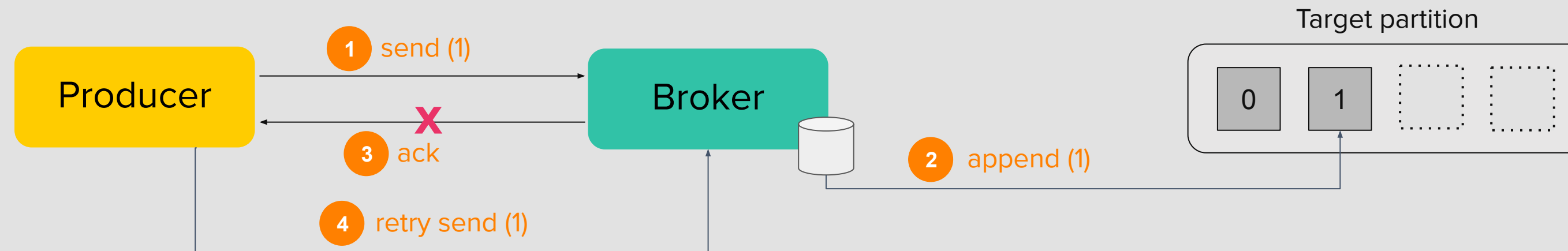


# Идемпотентные продюсеры

## Штатная работа



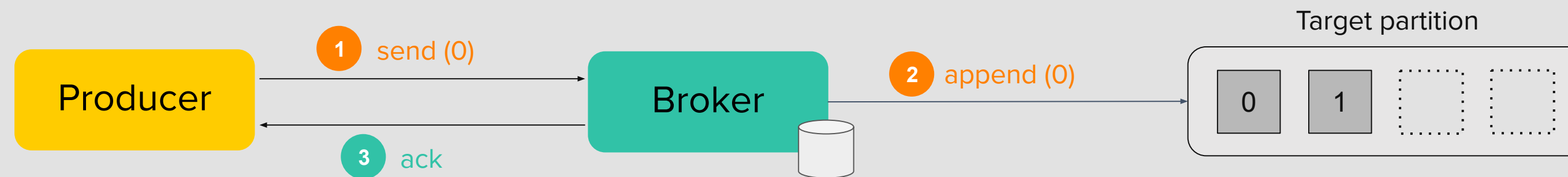
## Сбой при записи



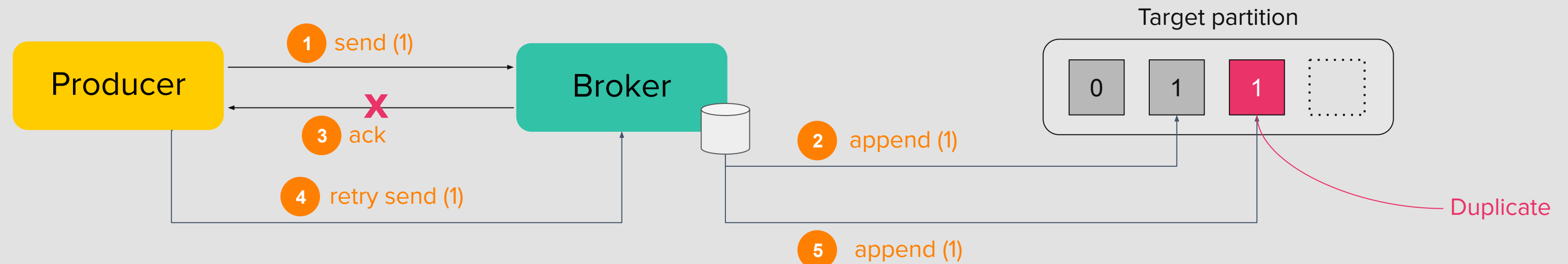


# Идемпотентные продюсеры

## Штатная работа



## Сбой при записи

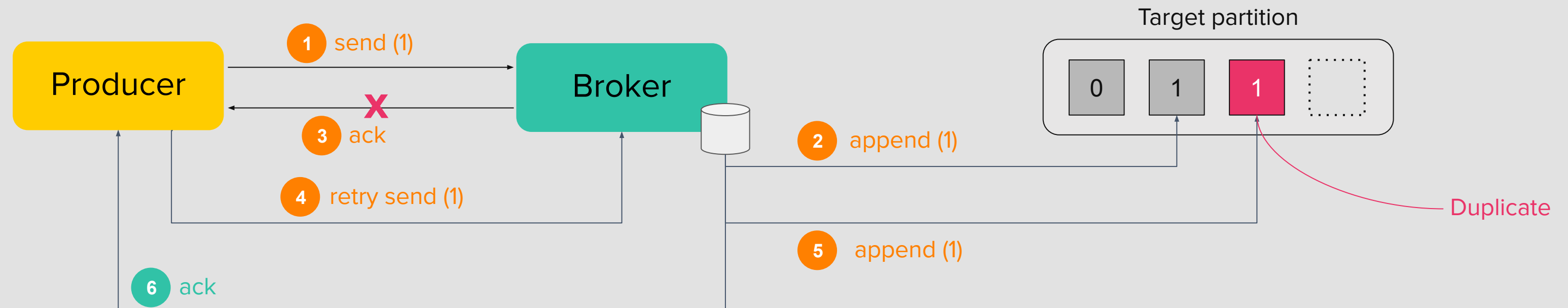


# Идемпотентные продюсеры

## Штатная работа



## Сбой при записи



# 01

Продюсеры  
партицируют данные в  
топиках

# 01

Продюсеры  
партицируют данные в  
топиках

# 02

Алгоритм  
партицирования  
определяется  
продюсером

# 01

Продюсеры  
партицируют данные в  
топиках

# 02

Алгоритм  
партицирования  
определяется  
продюсером

# 03

Продюсер определяет  
размер батча и число  
ретраев при отправке  
сообщений

# 01

Продюсеры  
партицируют данные в  
топиках

# 02

Алгоритм  
партицирования  
определяется  
продюсером

# 03

Продюсер определяет  
размер батча и число  
ретраев при отправке  
сообщений

# 04

Kafka предоставляет  
гарантии доставки  
at-most once, at-least once  
и exactly once

# 01

Продюсеры  
партицируют данные в  
топиках

# 02

Алгоритм  
партицирования  
определяется  
продюсером

# 03

Продюсер определяет  
размер батча и число  
ретраев при отправке  
сообщений

# 04

Kafka предоставляет  
гарантии доставки  
at-most once, at-least once  
и exactly once

# 05

Exactly-once нужен  
не каждому сервису

# 01

Продюсеры партицируют данные в топиках

# 02

Алгоритм партицирования определяется продюсером

# 03

Продюсер определяет размер батча и число ретраев при отправке сообщений

# 04

Kafka предоставляет гарантии доставки at-most once, at-least once и exactly once

# 05

Exactly-once нужен не каждому сервису

# 06

У exactly once есть цена: увеличение задержек при записи



# 01

Продюсеры  
партицируют данные в  
топиках

# 02

Алгоритм  
партицирования  
определяется  
продюсером

# 03

Продюсер определяет  
размер батча и число  
ретраев при отправке  
сообщений

# 04

Kafka предоставляет  
гарантии доставки  
at-most once, at-least once  
и exactly once

# 05

Exactly-once нужен  
не каждому сервису

# 06

У exactly once есть цена:  
увеличение задержек  
при записи

# 07

Указывайте  
`min.insync.replicas`  
меньше общего числа  
реплик

# 01

Продюсеры партицируют данные в топиках

# 02

Алгоритм партицирования определяется продюсером

# 03

Продюсер определяет размер батча и число ретраев при отправке сообщений

# 04

Kafka предоставляет гарантии доставки `at-most once`, `at-least once` и `exactly once`

# 05

`Exactly-once` нужен не каждому сервису

# 06

У `exactly once` есть цена: увеличение задержек при записи

# 07

Указывайте `min.insync.replicas` меньше общего числа реплик

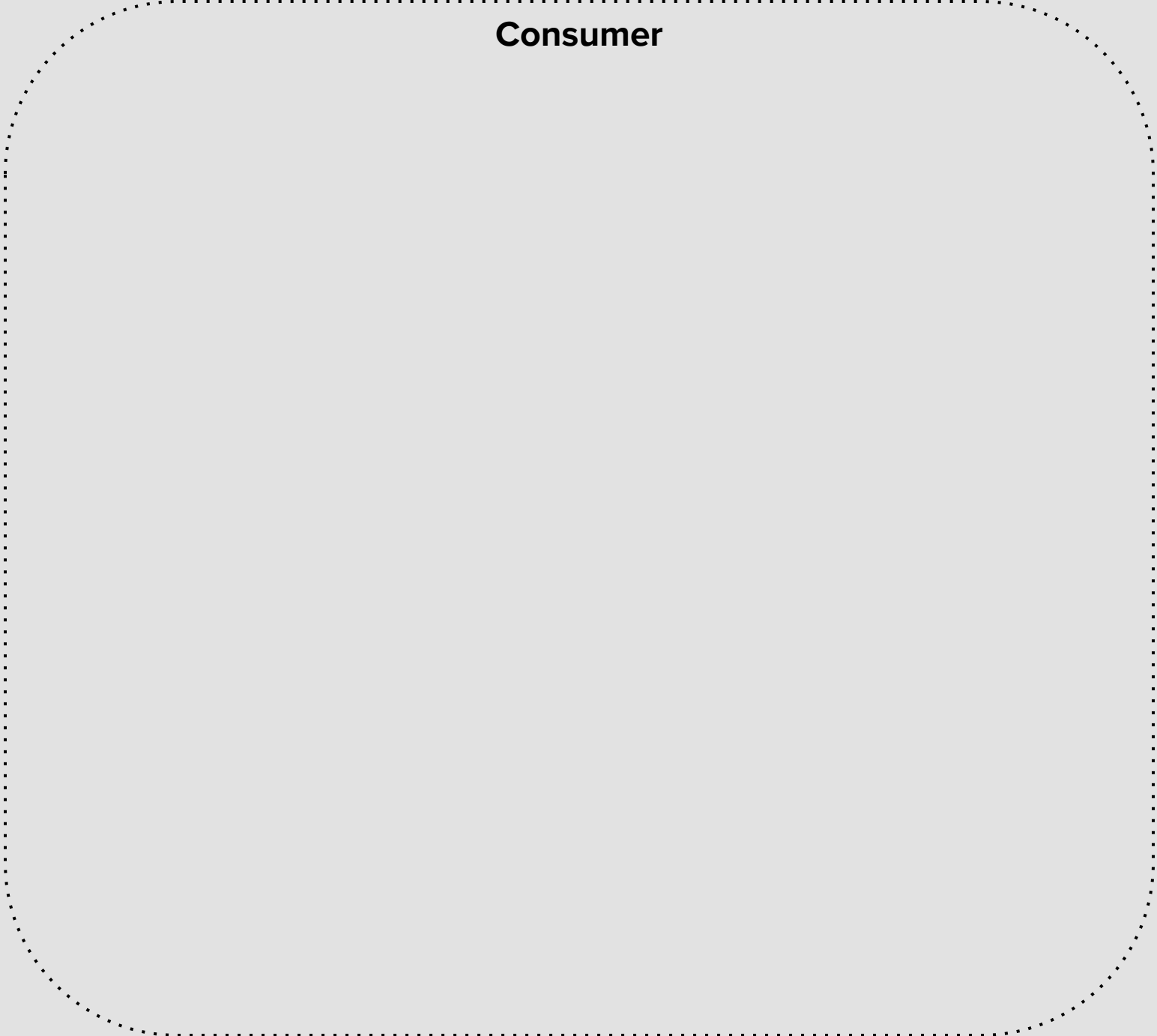
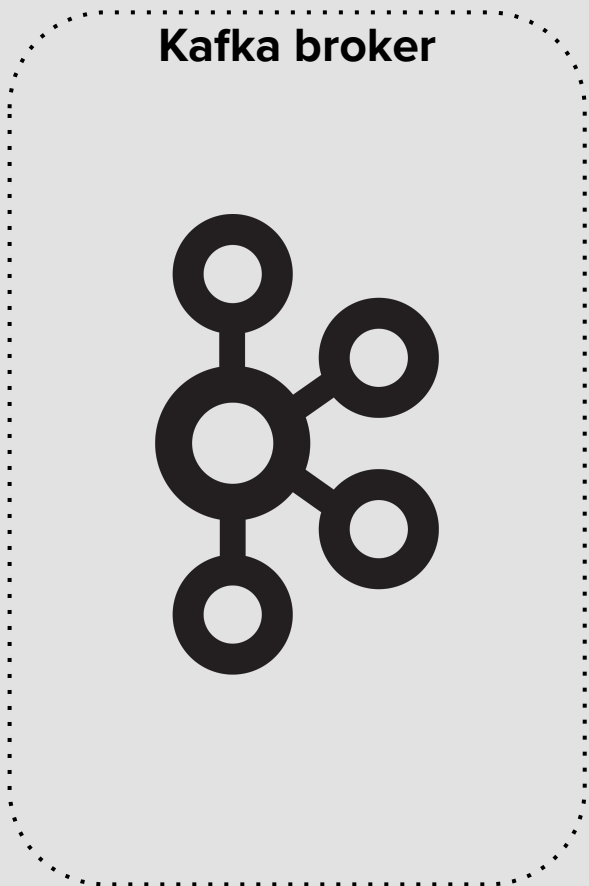
# 08

Если указываете `acks=all`, то включайте и `enable.idempotence`

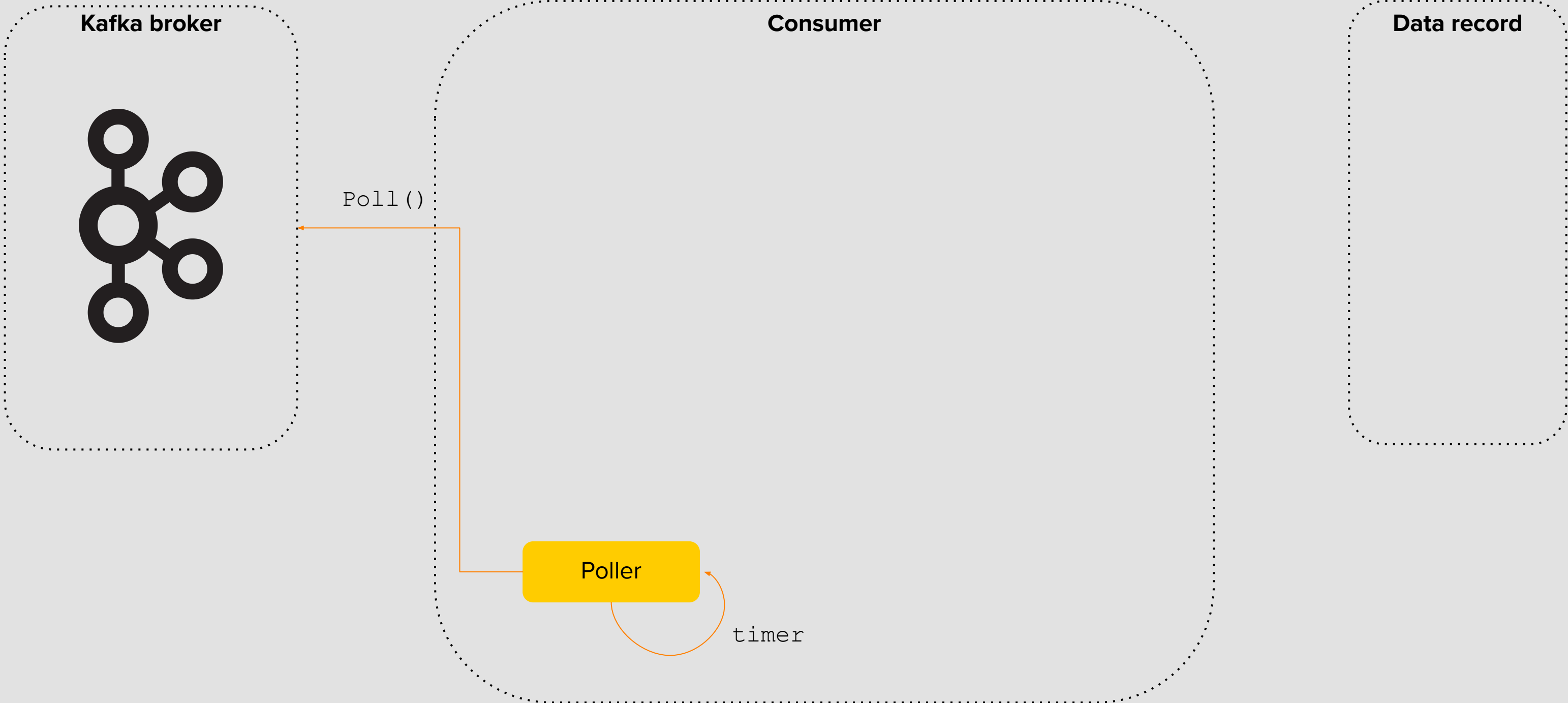
# Консумеры



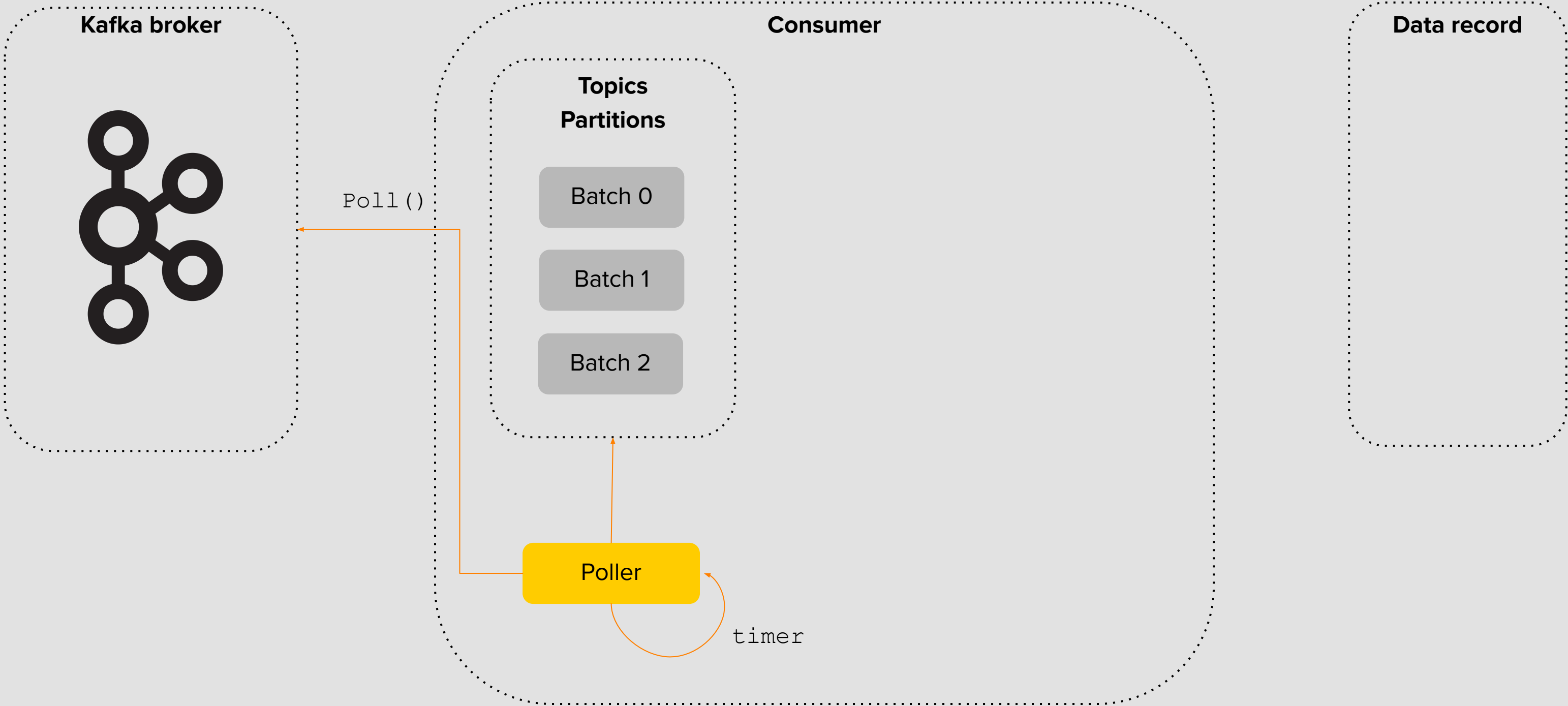
# Дизайн консумера



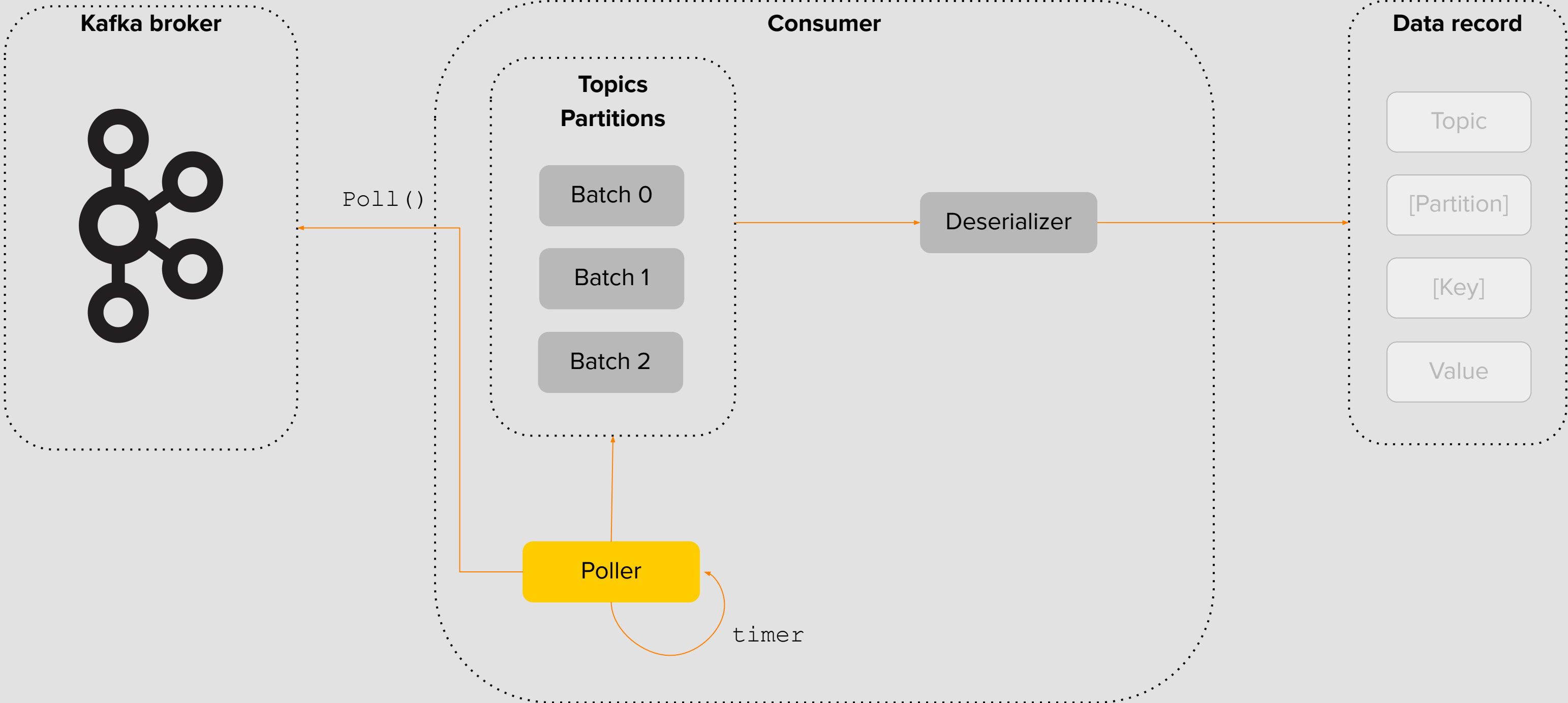
# Дизайн консумера



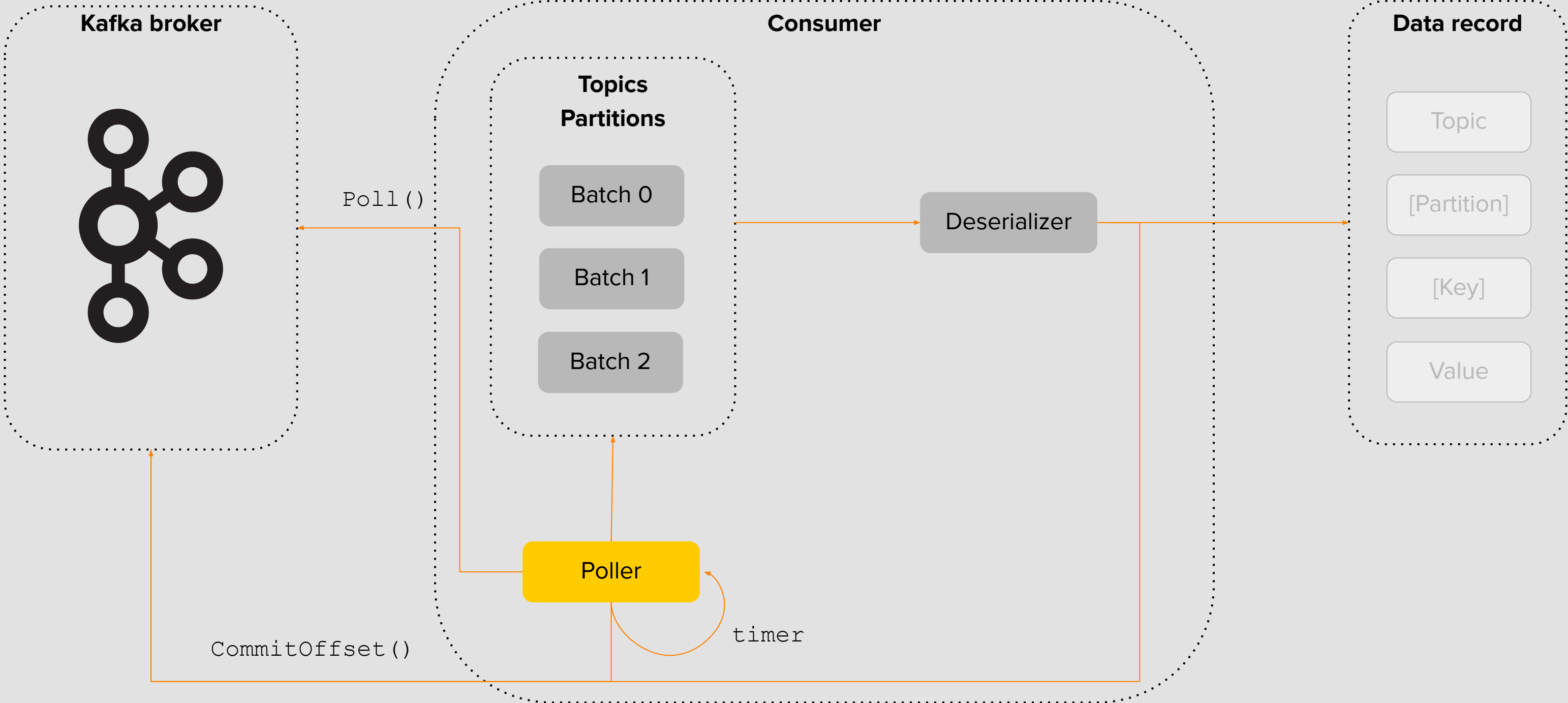
# Дизайн консумера



# Дизайн консумера

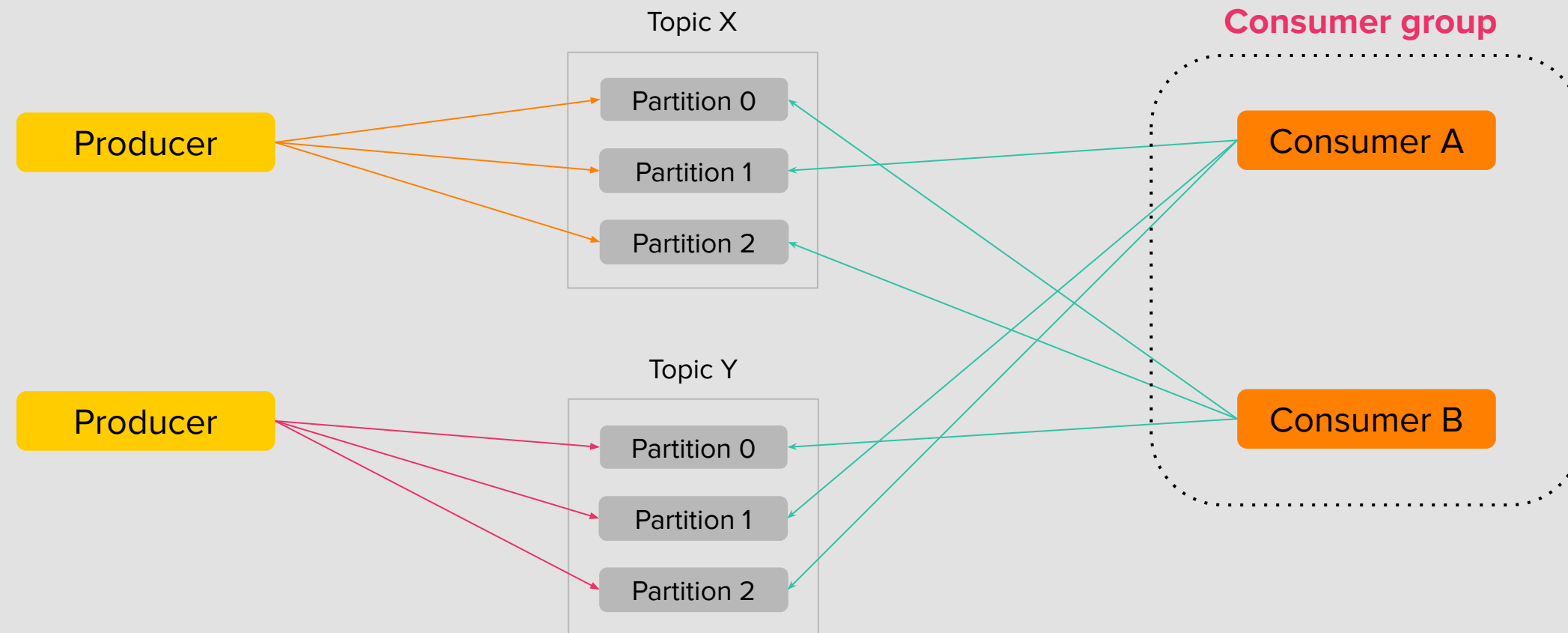


# Дизайн консумера

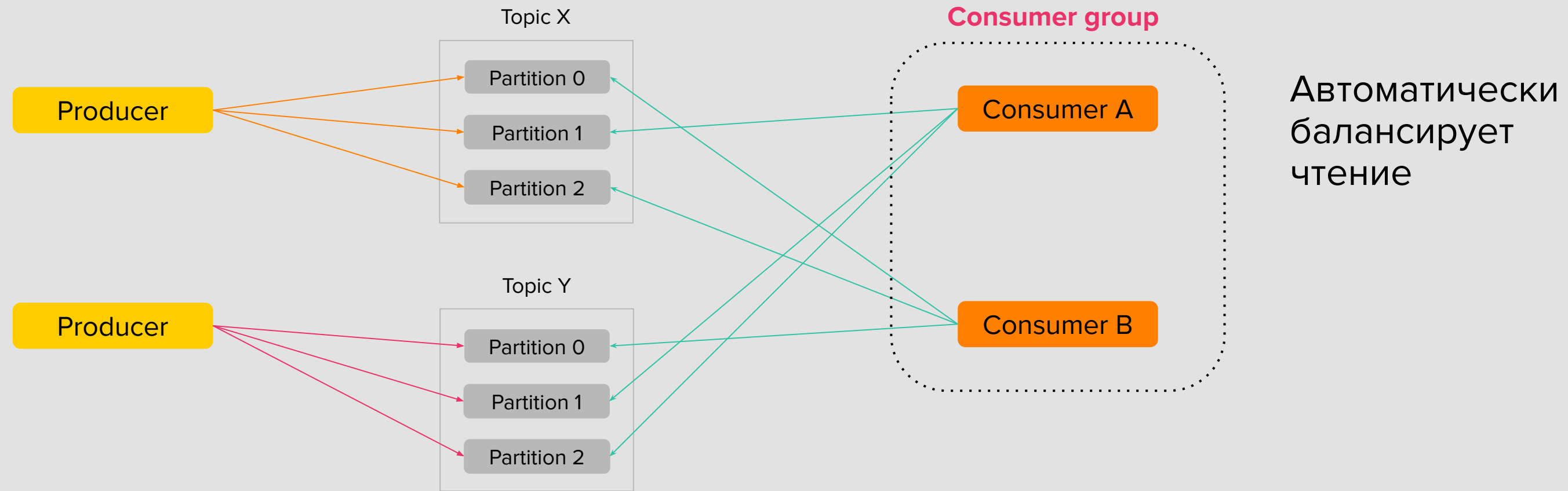




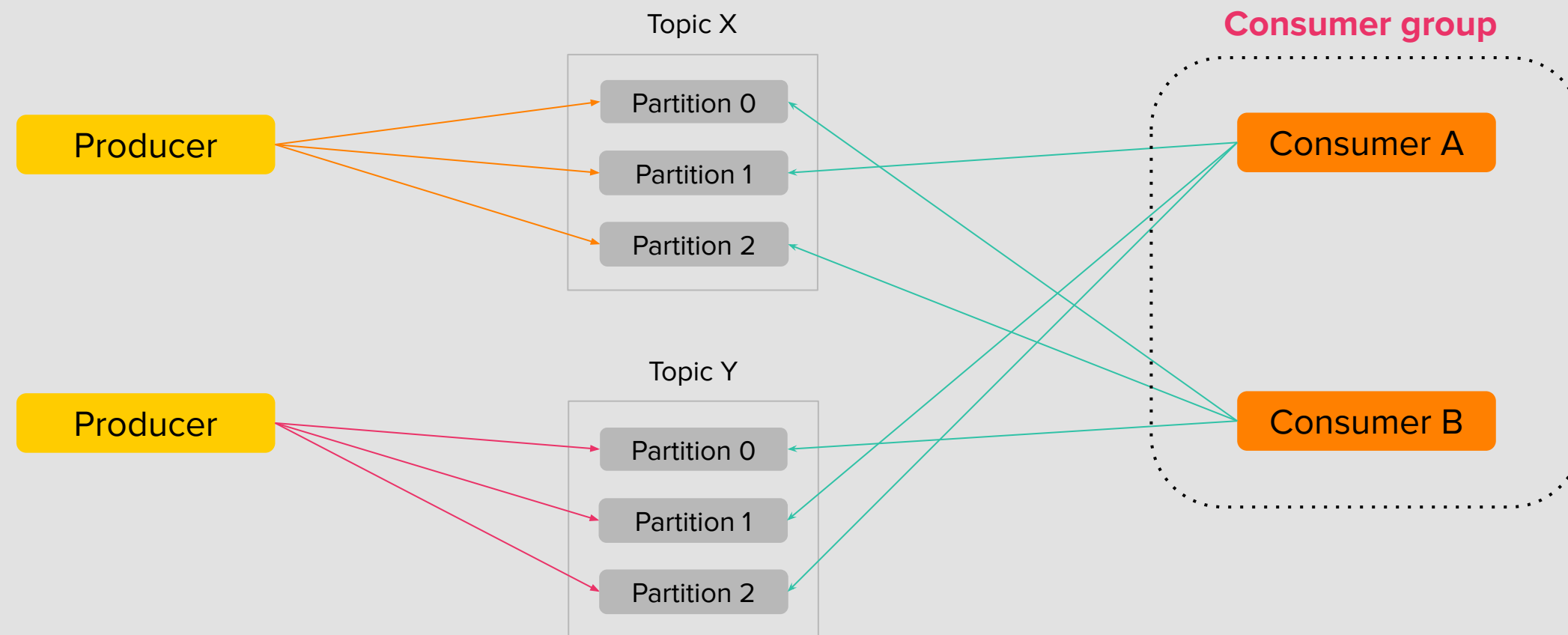
# Консумер-группы



# Консумер-группы



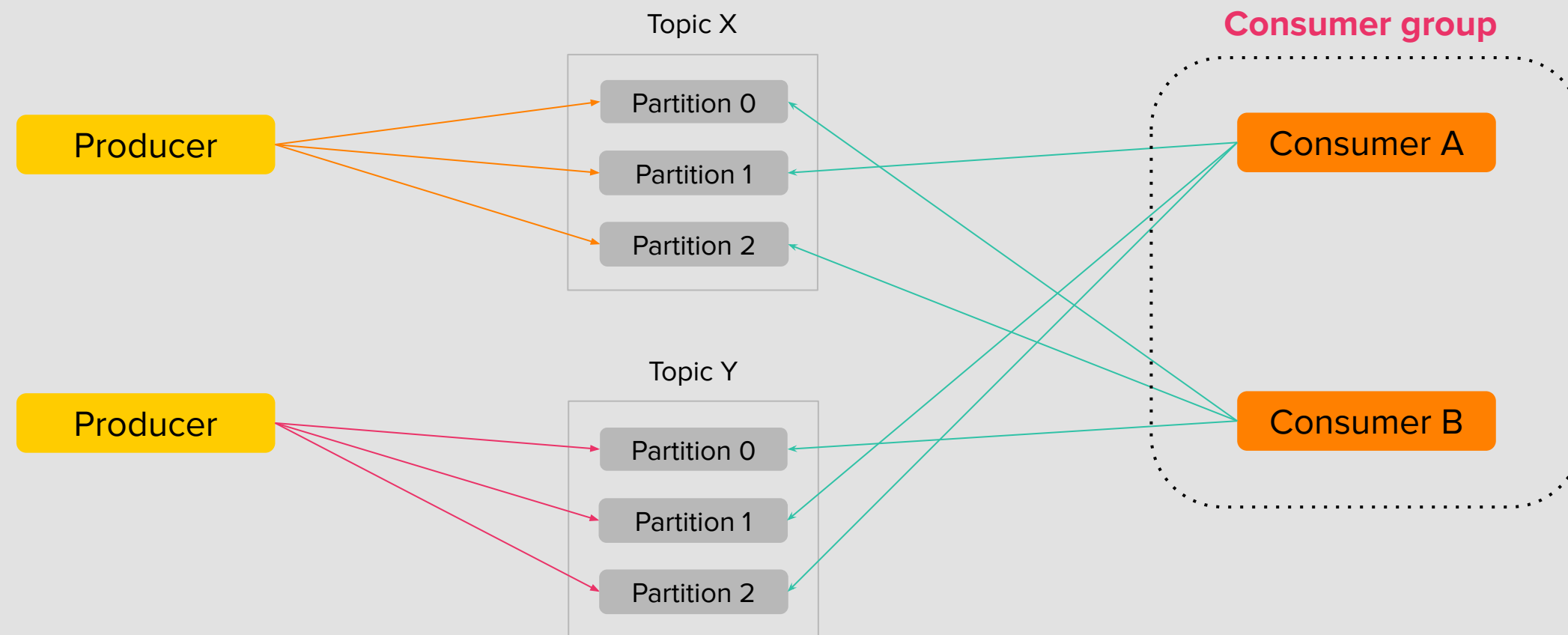
# Консумер-группы



Автоматически  
балансирует  
чтение

Имеет одинаковый  
group id

# Консумер-группы

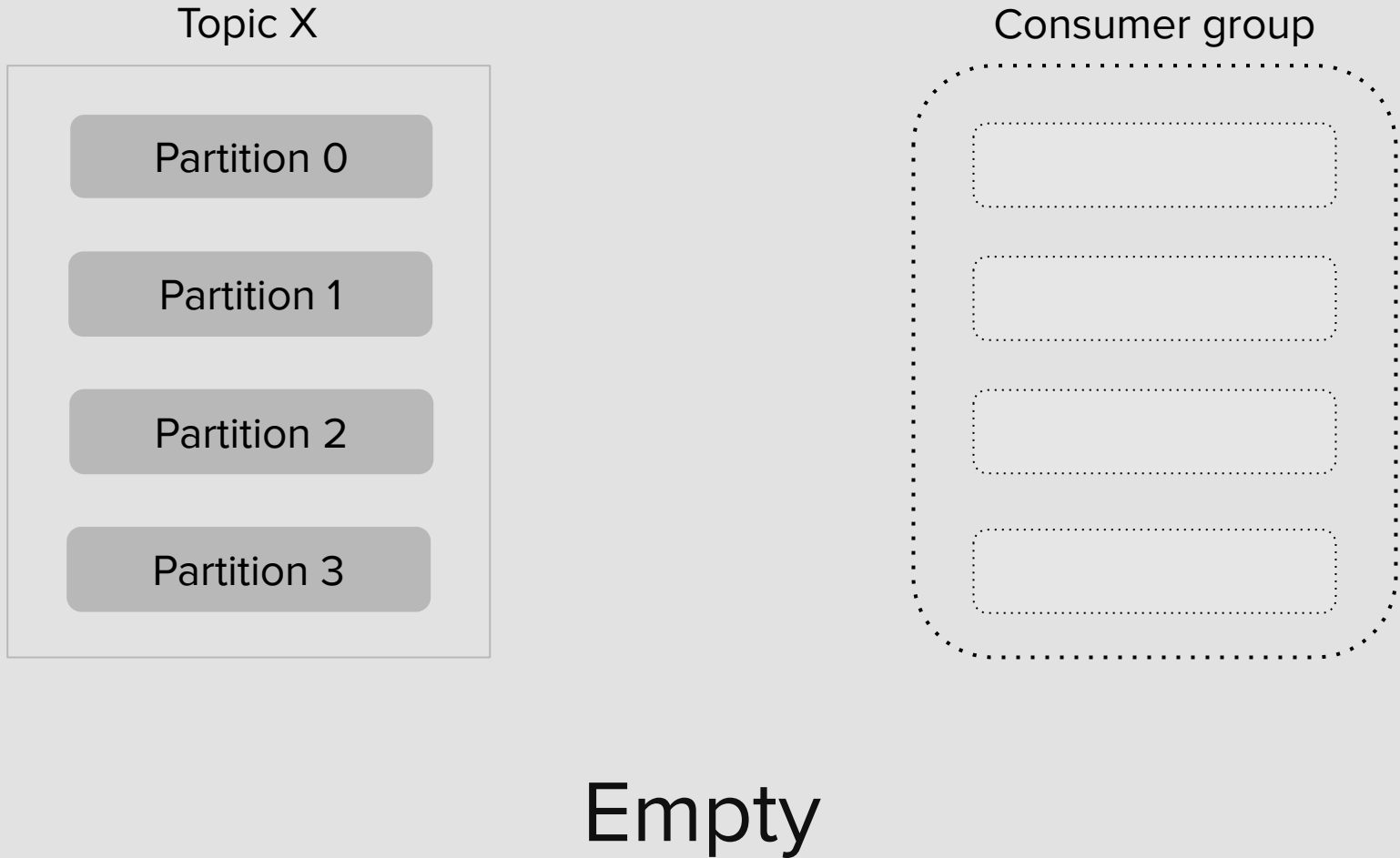


Автоматически  
балансирует  
чтение

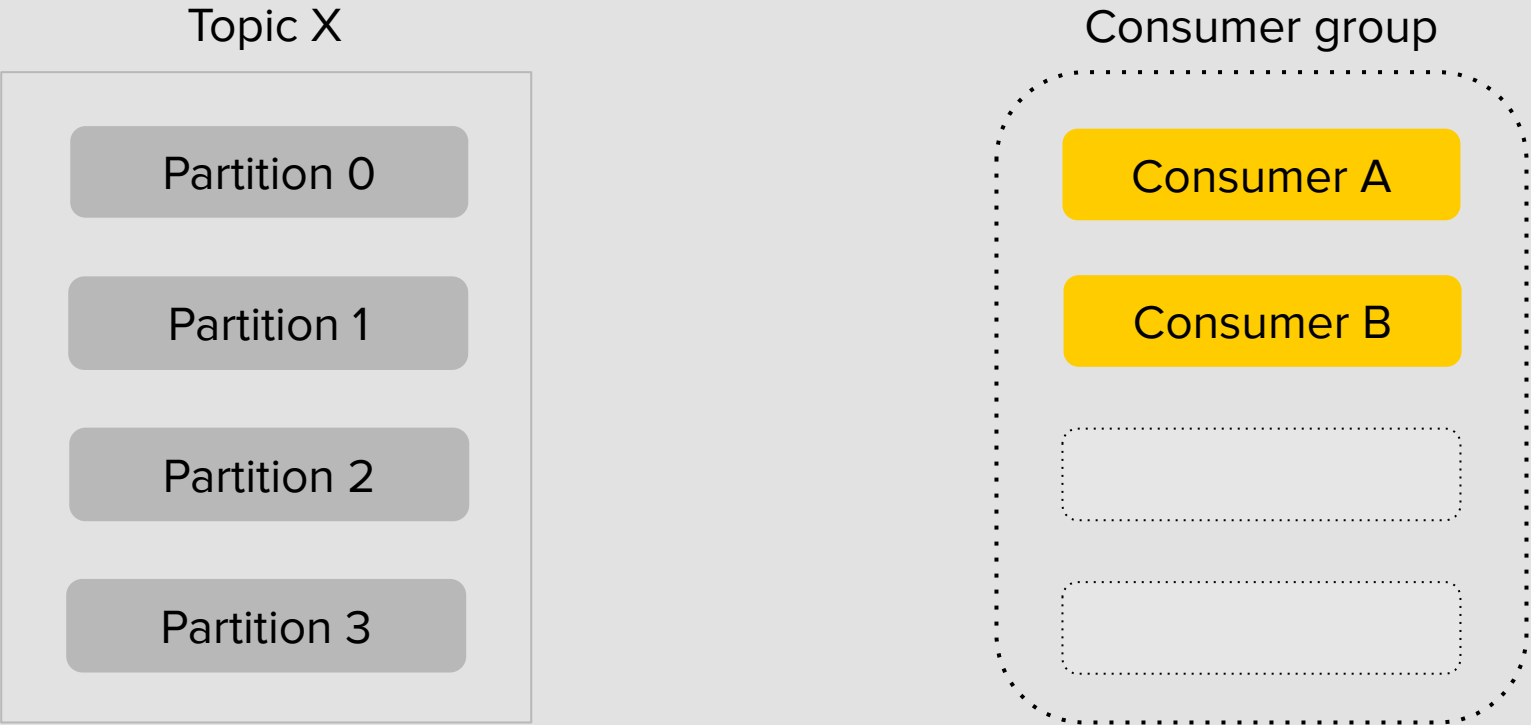
Имеет одинаковый  
group id

Хранит оффсеты  
на брокерах

# Рибалансировка консумер-групп

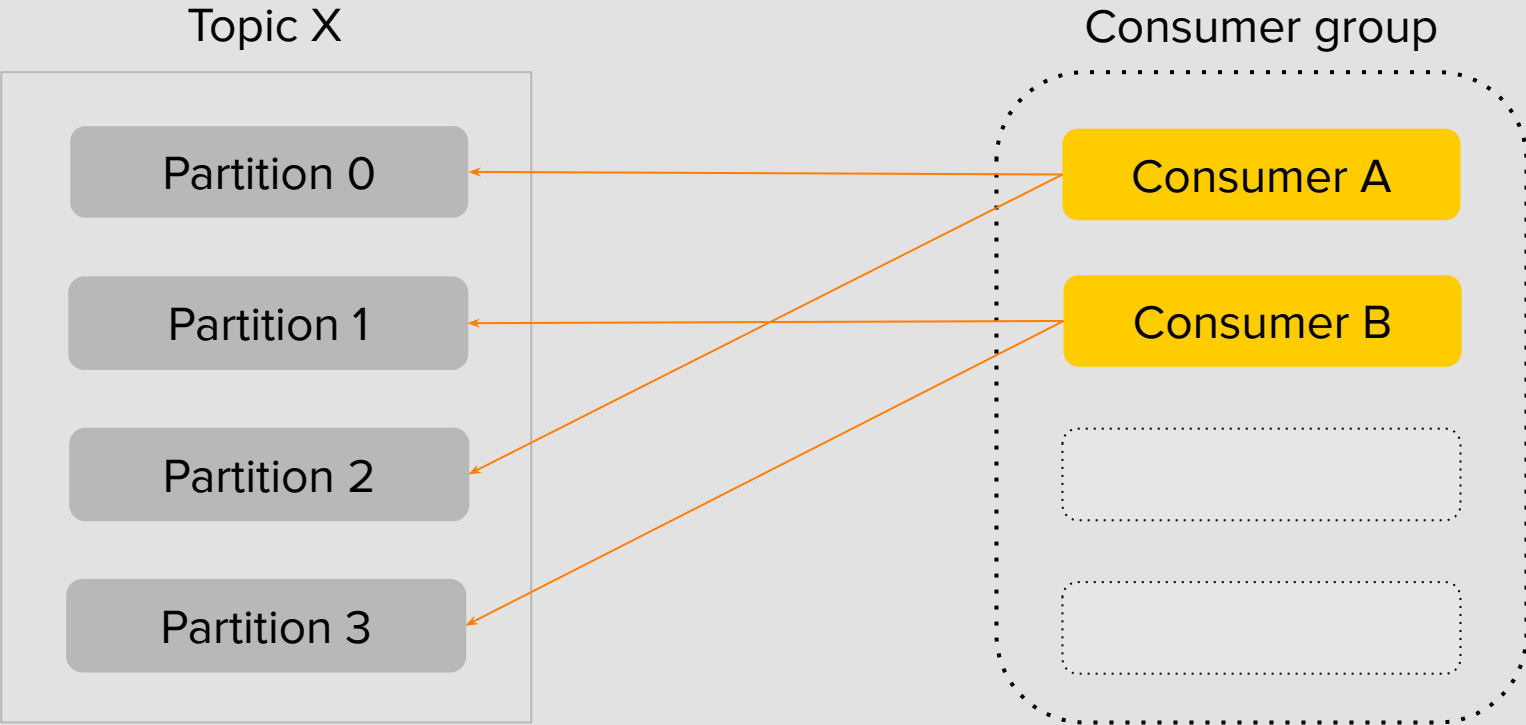


# Рибалансировка консумер-групп



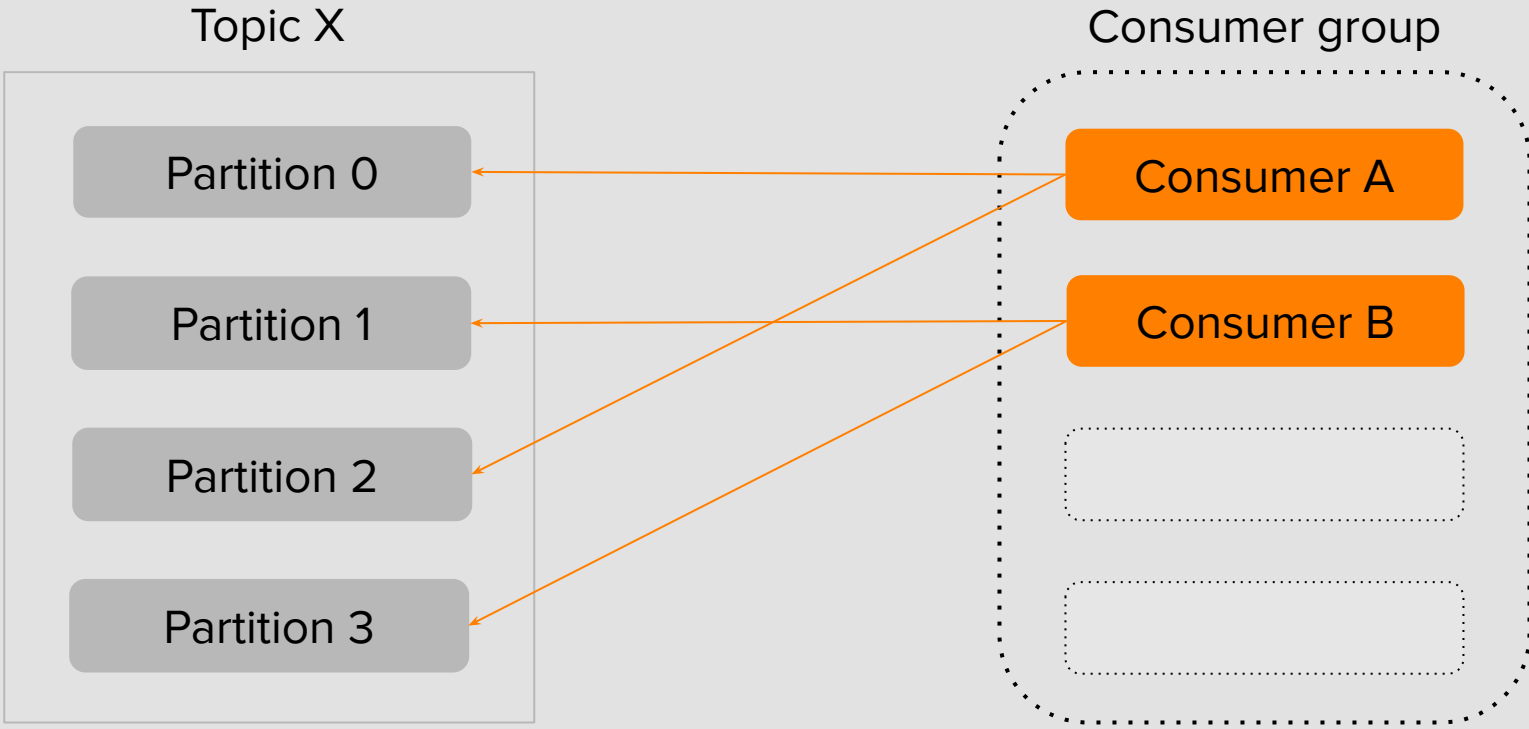
Rebalancing

# Рибалансировка консумер-групп



Rebalancing

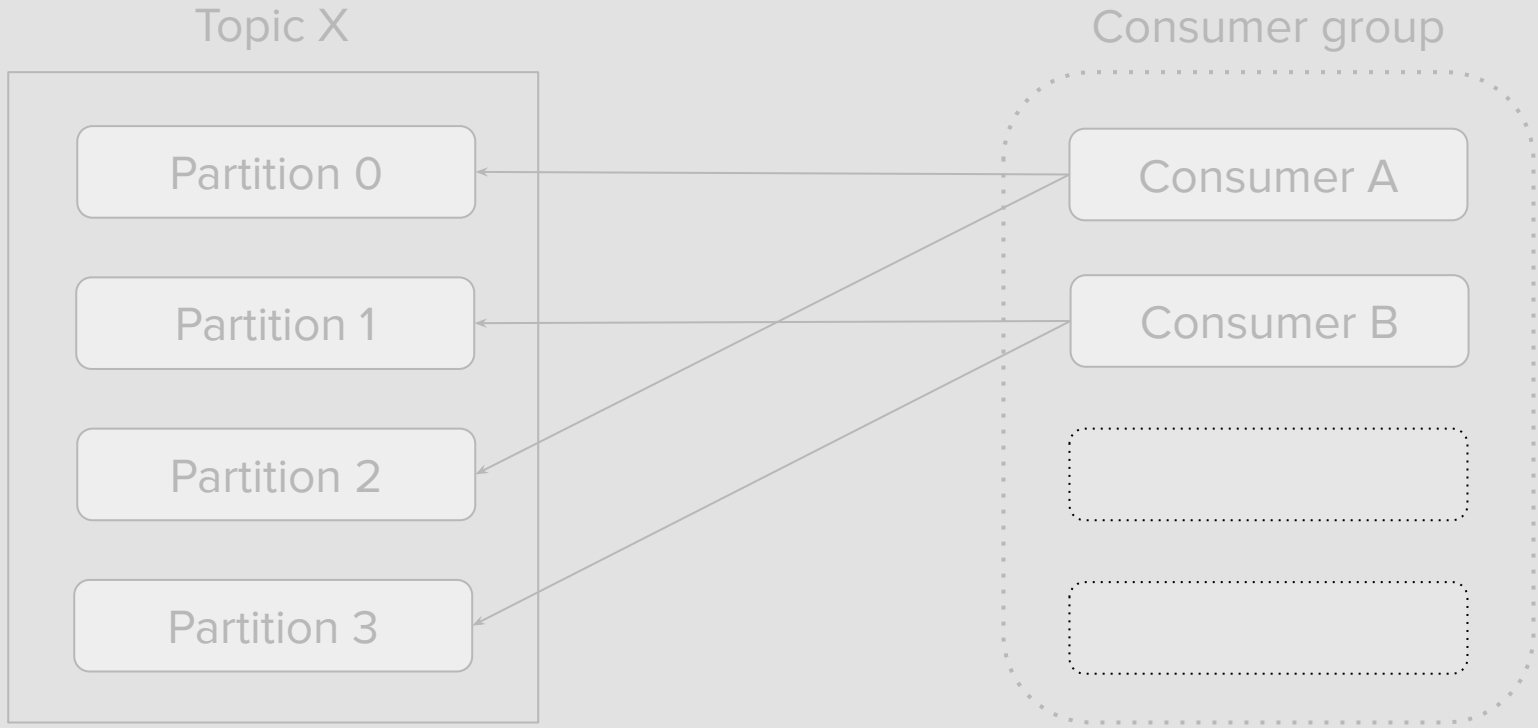
# Рибалансировка консумер-групп



Stable



# Рибалансировка консумер-групп



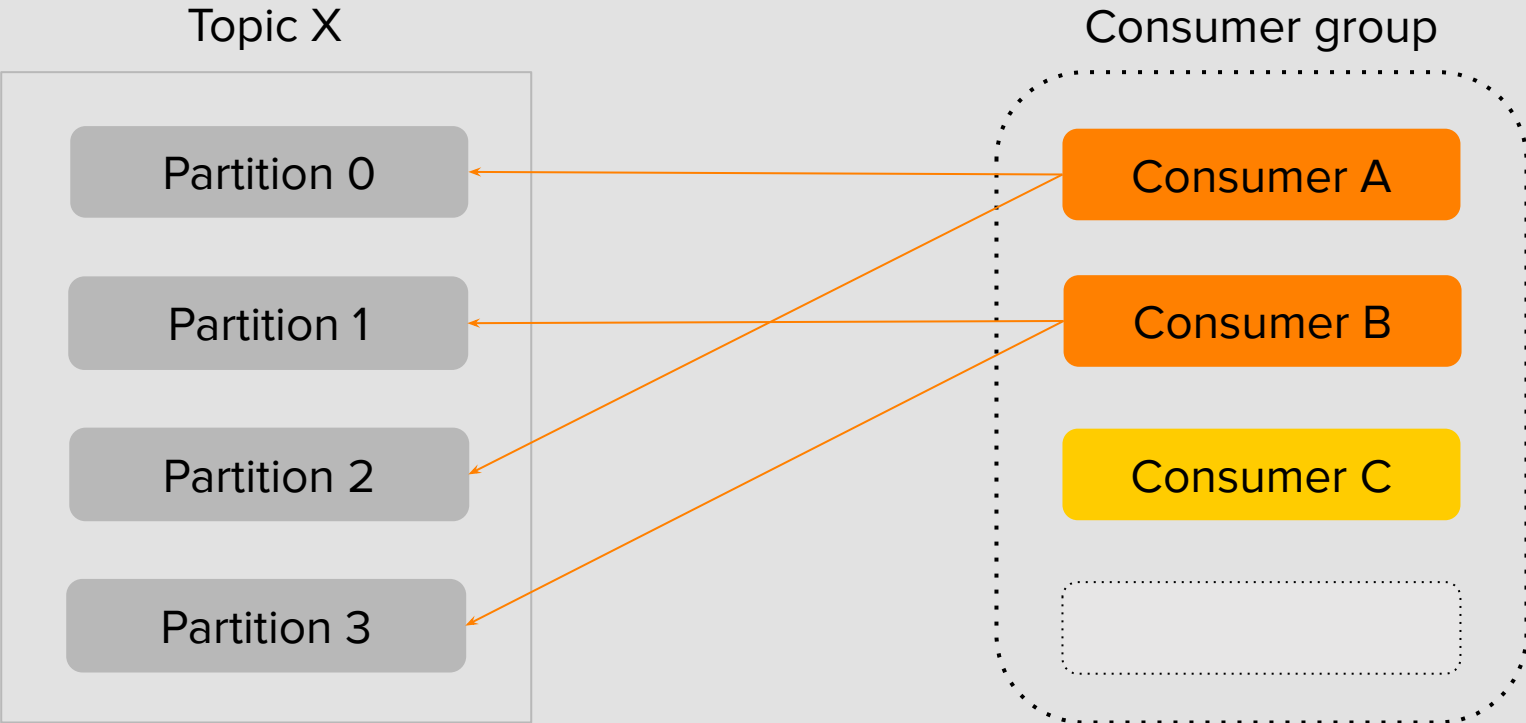
earliest

auto.offset.reset

latest

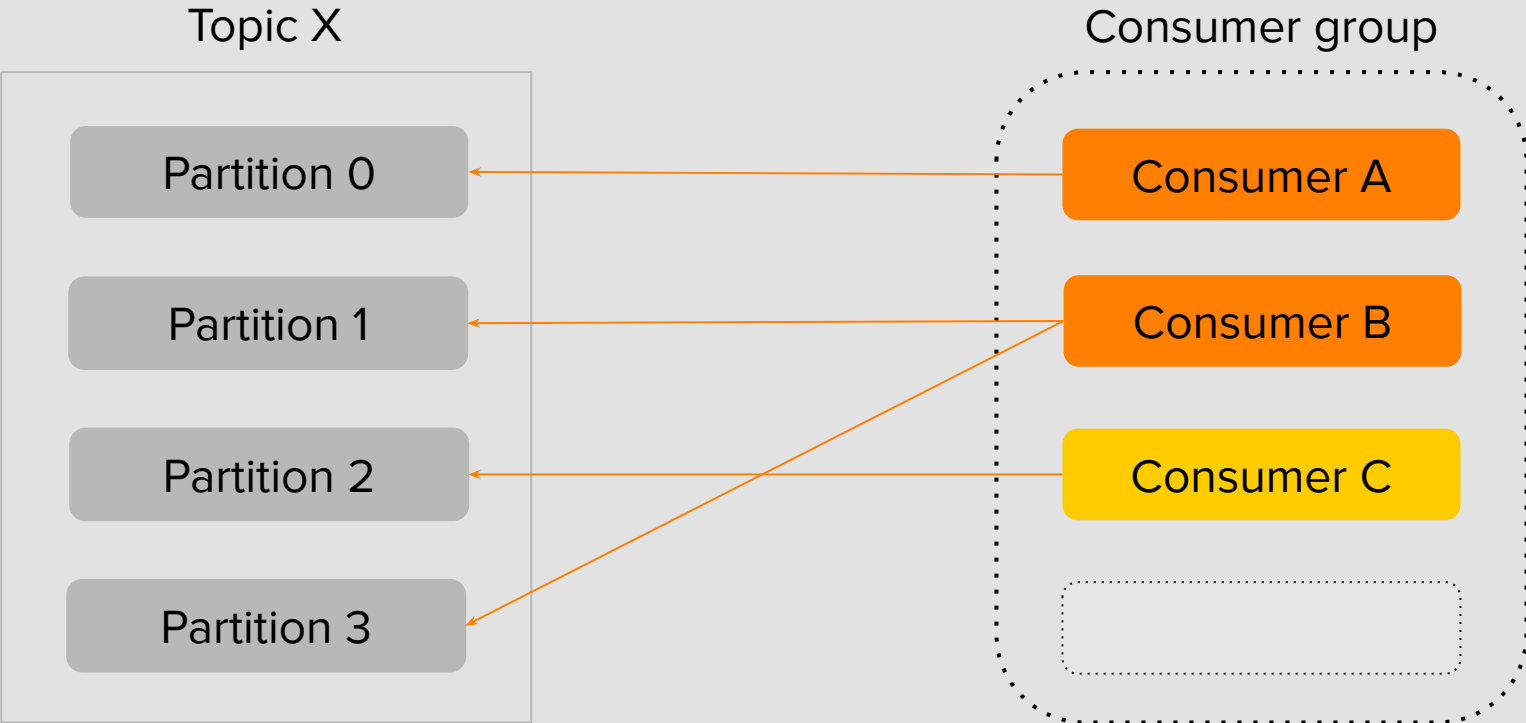


# Рибалансировка консумер-групп



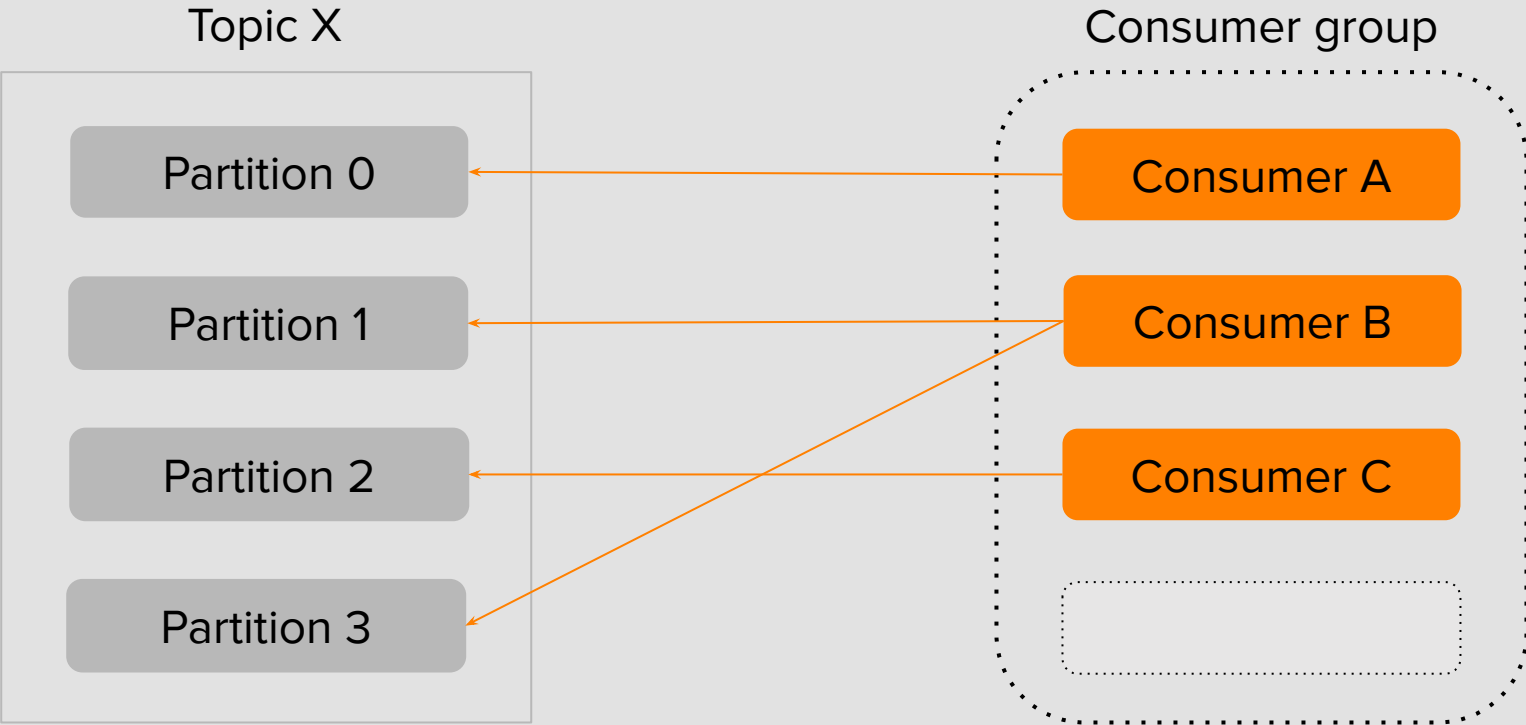
Rebalancing

# Рибалансировка консумер-групп



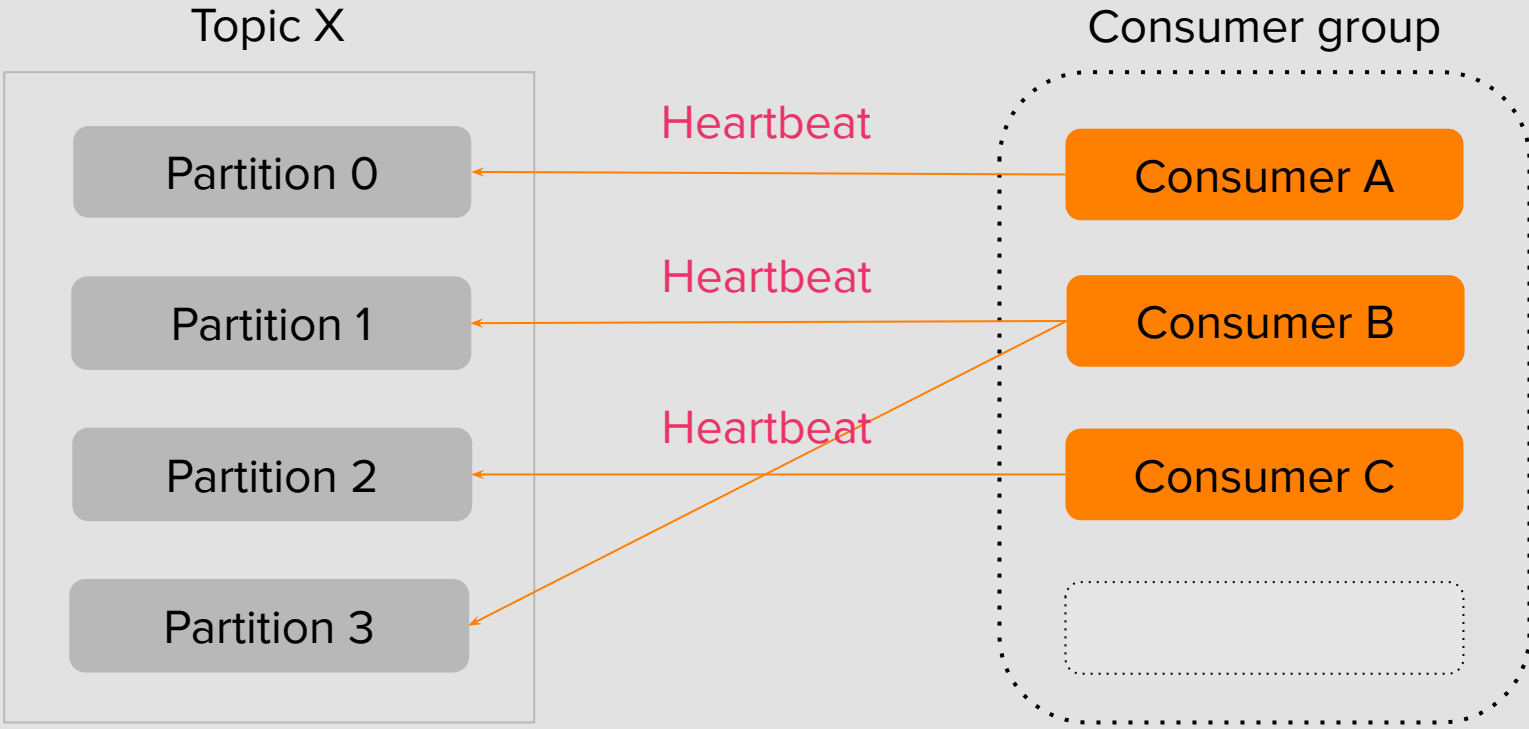
Rebalancing

# Рибалансировка консумер-групп



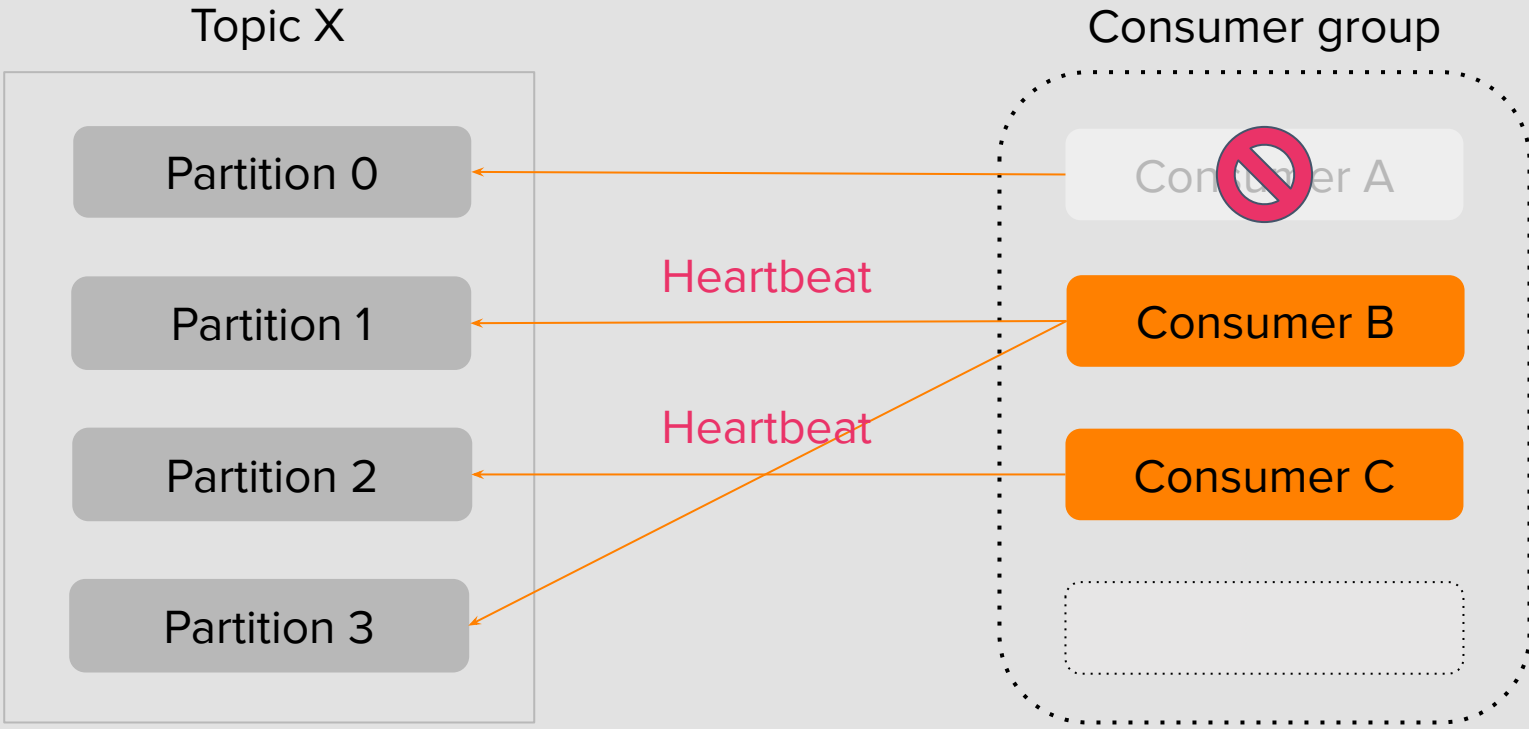
Stable

# Рибалансировка консумер-групп



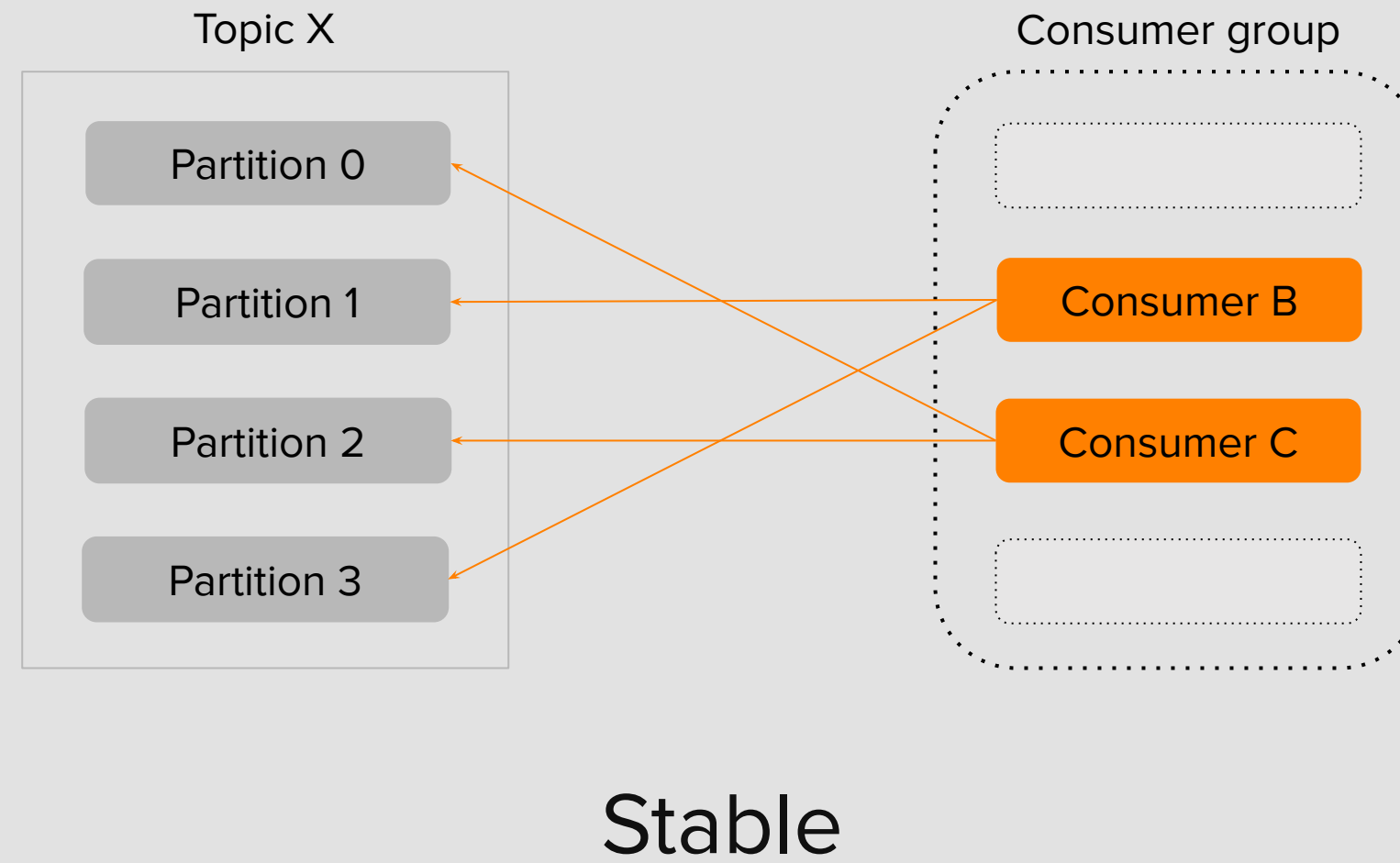
Stable

# Рибалансировка консумер-групп

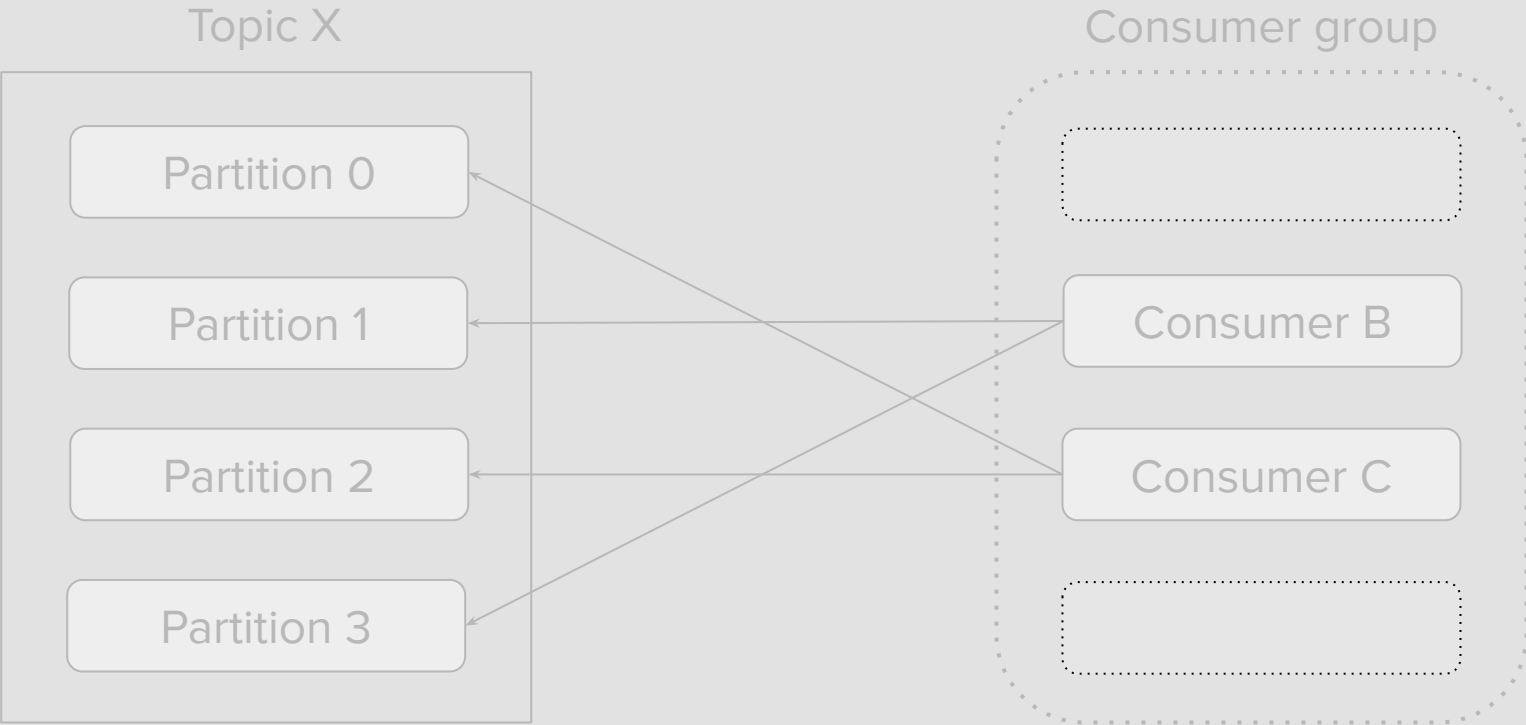


Rebalancing

# Рибалансировка консумер-групп



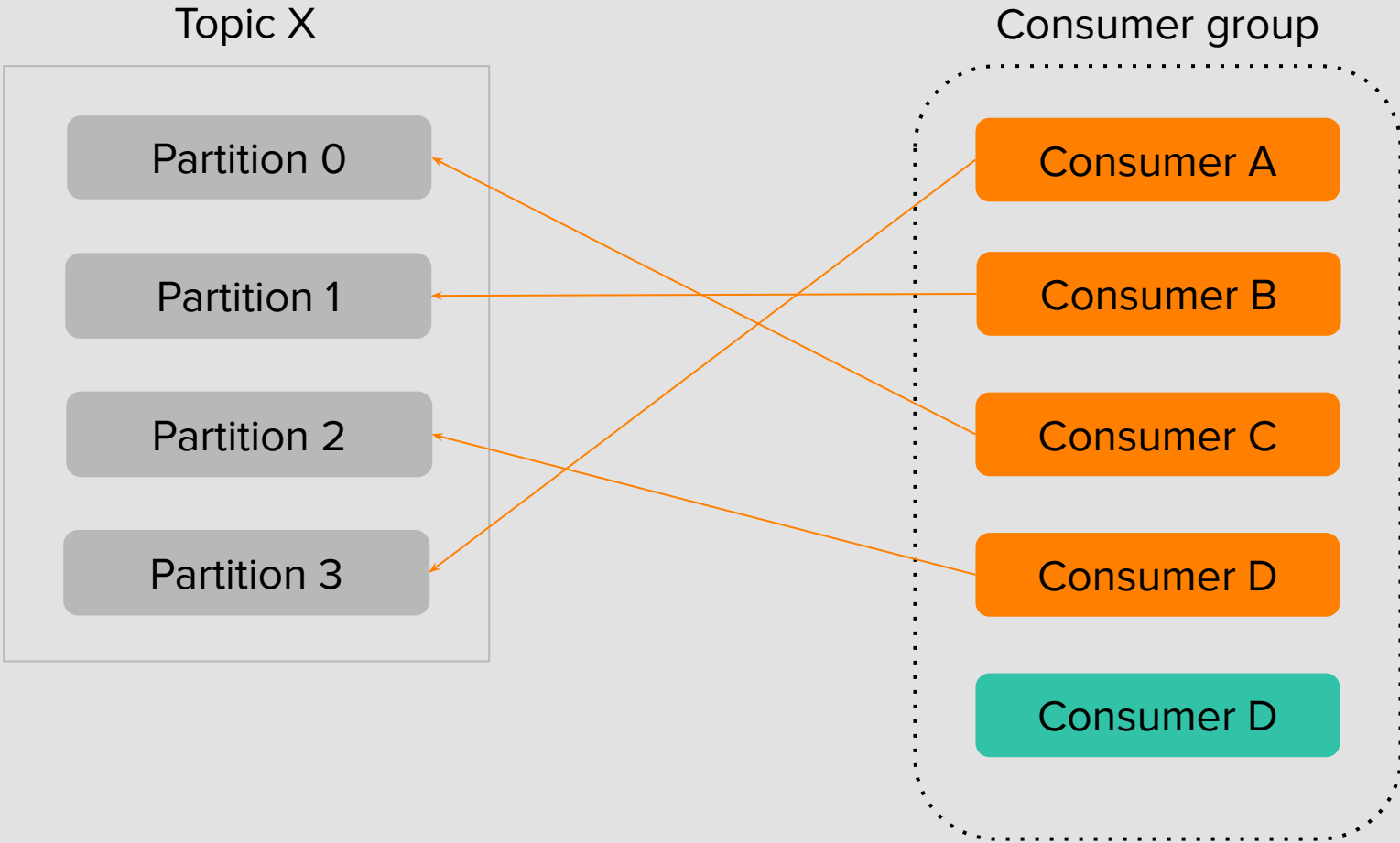
# Рибалансировка консумер-групп



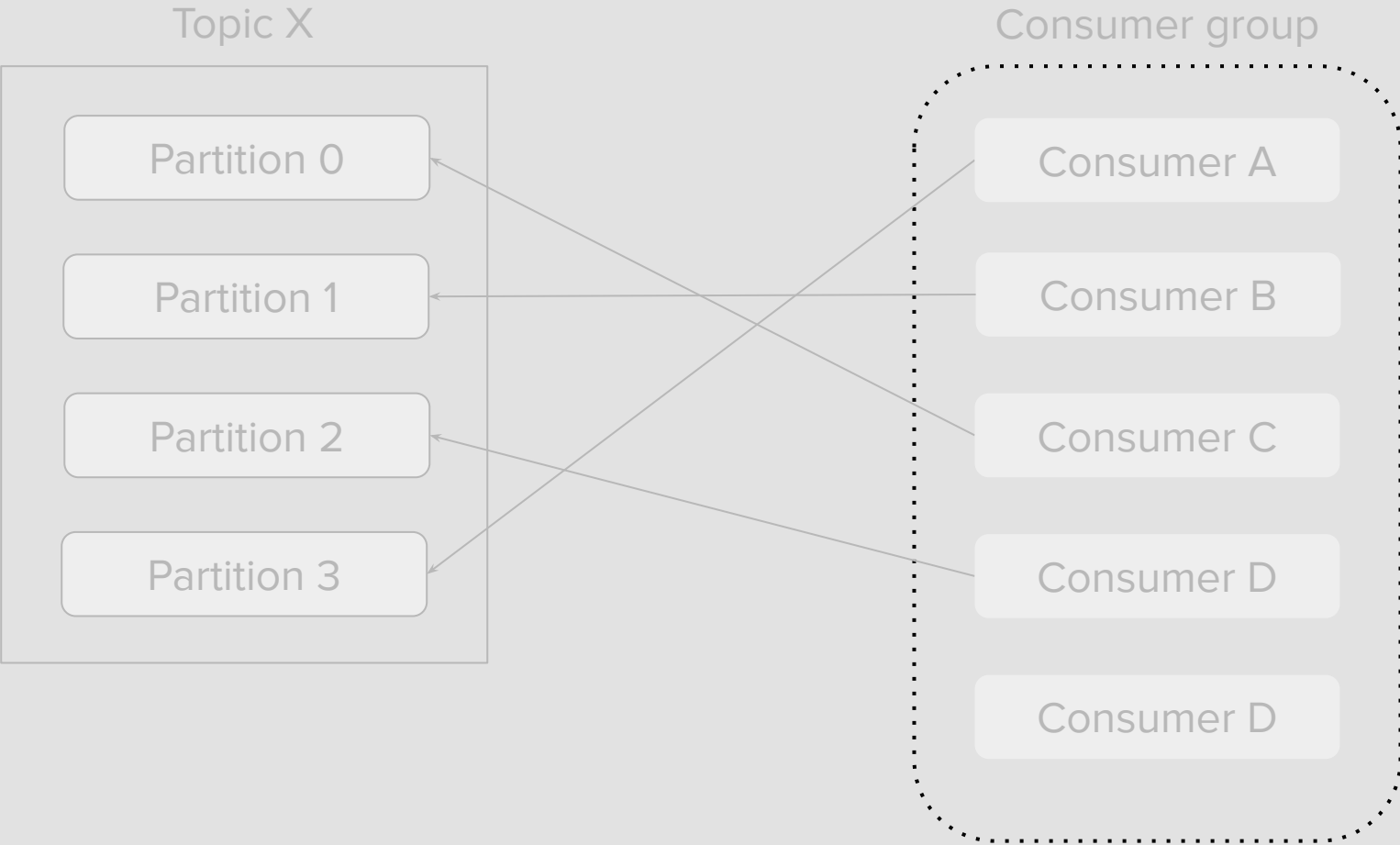
3 x heartbeat.interval.ms < session.timeout.ms



# Рибалансировка консумер-групп



# Рибалансировка консумер-групп



Консумеров в группе не больше партиций в топике

# 01

Партиции в консумер-  
группах распределяются  
автоматически

# 01

Партиции в консумер-группах распределяются автоматически

# 02

Один консумер в группе может читать одну и более партиций

# 01

Партиции в консумер-группах распределяются автоматически

# 02

Один консумер в группе может читать одну и более партиций

# 03

Брокер хранит оффсеты партиций в консумер-группах

# 01

Партиции в консумер-группах распределяются автоматически

# 02

Один консумер в группе может читать одну и более партиций

# 03

Брокер хранит оффсеты партиций в консумер-группах

# 04

Для проверки живости консумеров в группе, консумеры отправляют Heartbeat-сообщения

# 01

Партиции в консумер-группах распределяются автоматически

# 02

Один консумер в группе может читать одну и более партиций

# 03

Брокер хранит оффсеты партиций в консумер-группах

# 04

Для проверки живости консумеров в группе, консумеры отправляют Heartbeat-сообщения

# 05

Смена композиции в консумер-группе запускает ребалансировку

# 01

Партиции в консумер-группах распределяются автоматически

# 02

Один консумер в группе может читать одну и более партиций

# 03

Брокер хранит оффсеты партиций в консумер-группах

# 04

Для проверки живости консумеров в группе, консумеры отправляют Heartbeat-сообщения

# 05

Смена композиции в консумер-группе запускает ребалансировку

# 06

Ребалансировка означает остановку всех консумеров до стабилизации



# 01

Партиции в консумер-группах распределяются автоматически

# 02

Один консумер в группе может читать одну и более партиций

# 03

Брокер хранит оффсеты партиций в консумер-группах

# 04

Для проверки живости консумеров в группе, консумеры отправляют Heartbeat-сообщения

# 05

Смена композиции в консумер-группе запускает ребалансировку

# 06

Ребалансировка означает остановку всех консумеров до стабилизации

# 07

В новом консумере важно выбрать политику оффсета: читать с начала или с конца

# 01

Партиции в консумер-группах распределяются автоматически

# 02

Один консумер в группе может читать одну и более партиций

# 03

Брокер хранит оффсеты партиций в консумер-группах

# 04

Для проверки живости консумеров в группе, консумеры отправляют Heartbeat-сообщения

# 05

Смена композиции в консумер-группе запускает ребалансировку

# 06

Ребалансировка означает остановку всех консумеров до стабилизации

# 07

В новом консумере важно выбрать политику оффсета: читать с начала или с конца

# 08

Два и более консумеров в группе не могут читать из одной и той же партиции

# Вопросы



[github.com/gongled/kafka-workshop](https://github.com/gongled/kafka-workshop)