



ИНСТИТУТ
ДОПОЛНИТЕЛЬНОГО
ОБРАЗОВАНИЯ
УНИВЕРСИТЕТА ИННОПОЛИС

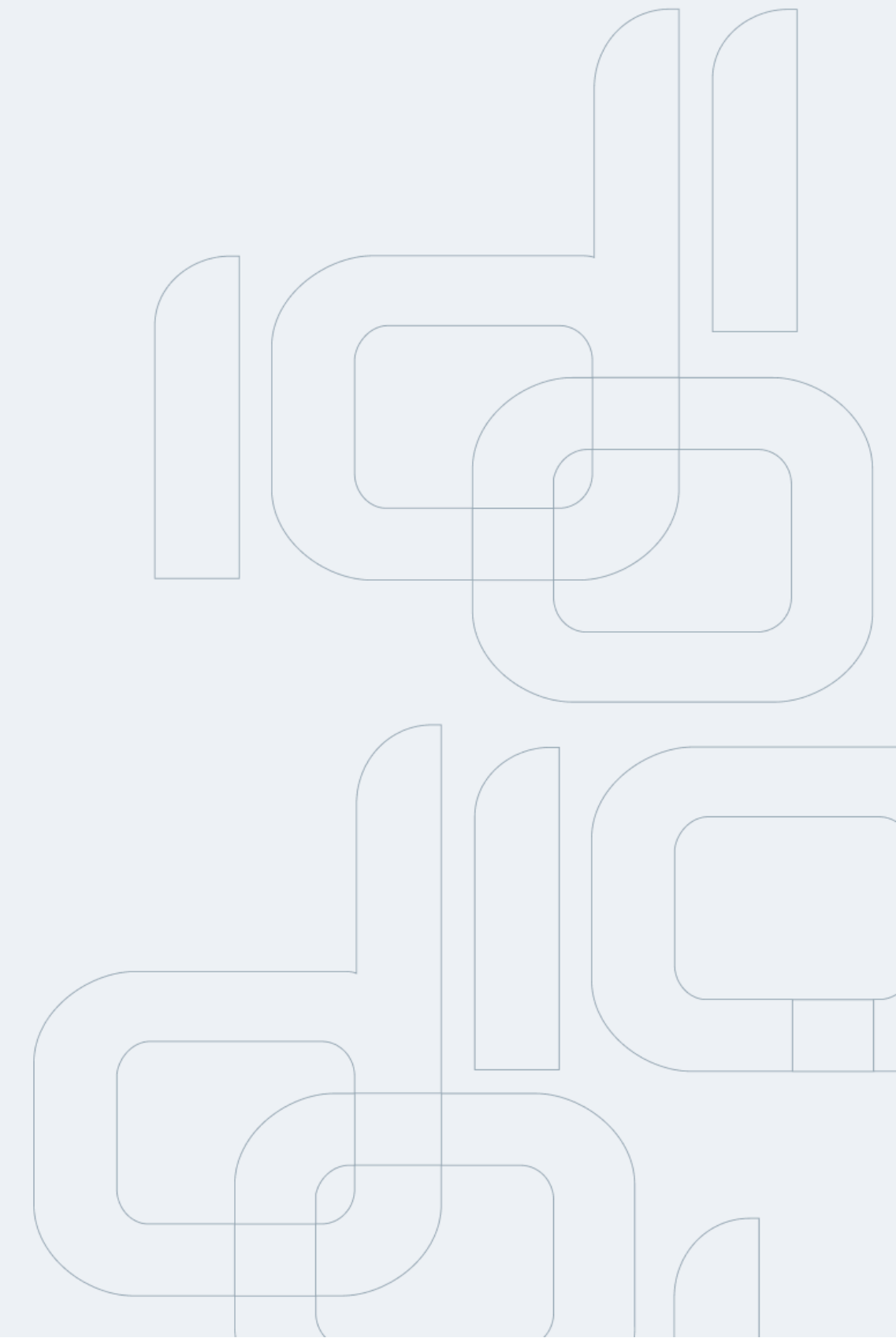
Потоковый сбор данных

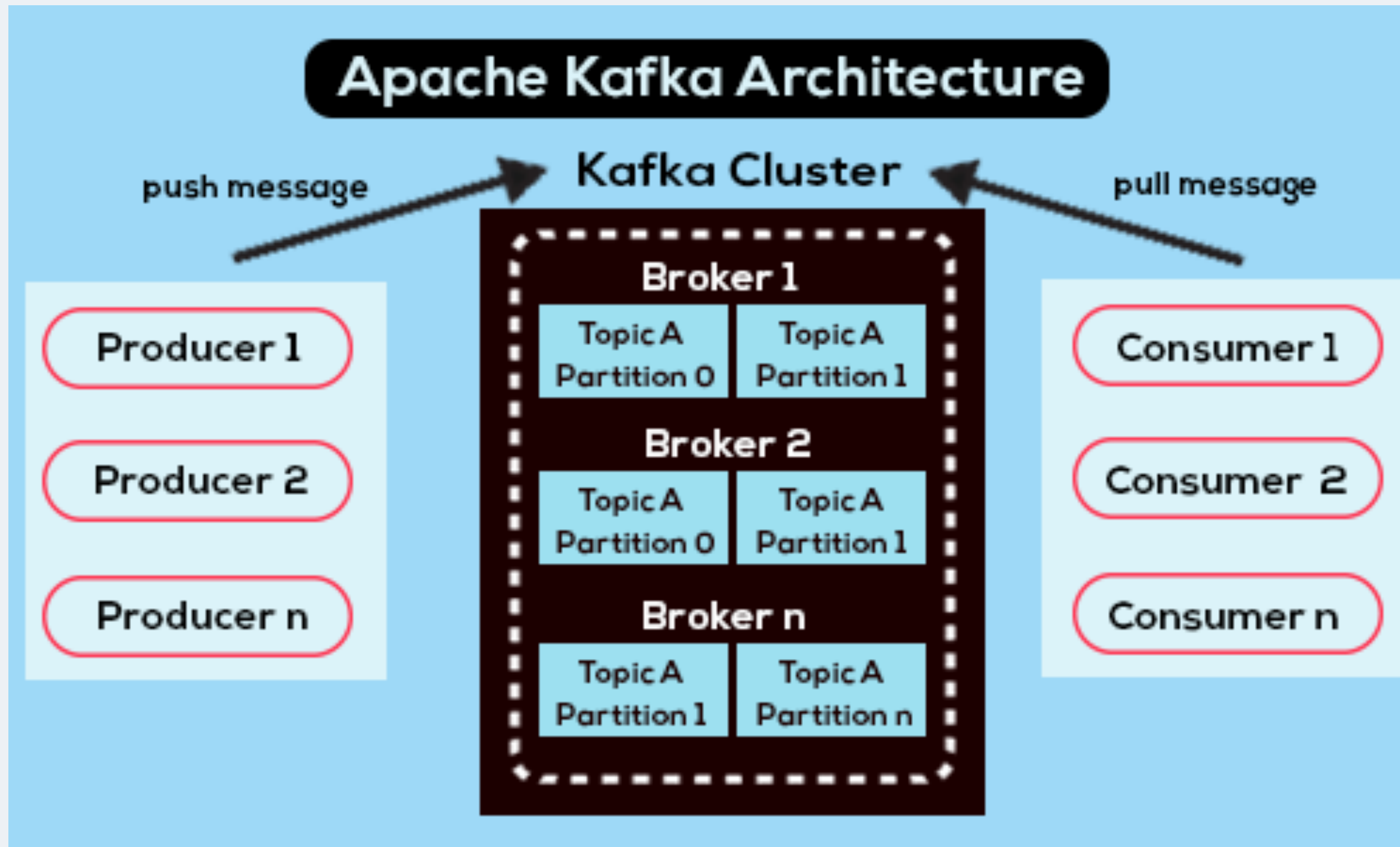
Мухамадуллин Ильяс
ilyasmukh@icloud.com



Что такое Kafka

Apache Kafka - это распределенная платформа для потоковой обработки и передачи данных.





1.Producer (Производитель):

1. Producer в Kafka - это компонент, который отправляет (публикует) сообщения (события) в топики Kafka. Производители ответственны за размещение данных в брокере Kafka.

2.Consumer (Потребитель):

1. Consumer в Kafka - это компонент, который читает (подписывается на) сообщения из топиков Kafka. Потребители обрабатывают данные, отправленные производителями, и могут реагировать на события в реальном времени.

3.Topic (Топик):

1. Топик в Kafka - это категоризированное место для хранения сообщений. Он представляет собой именованный канал, через который производители отправляют данные, и потребители читают данные. Топики могут быть логически разделены на разные категории данных.

4.Broker (Брокер):

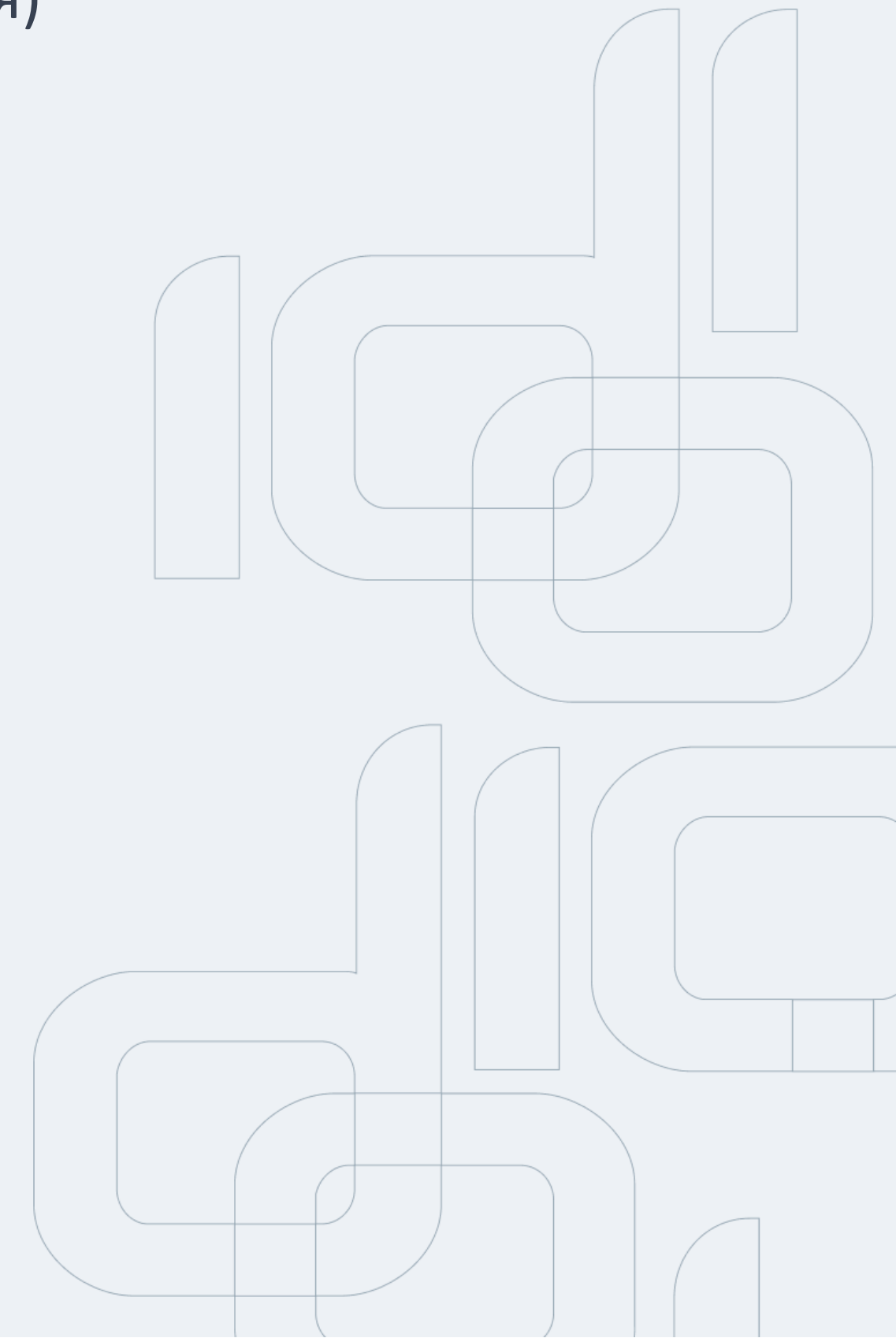
1. Брокер в Kafka - это сервер, на котором хранятся топики и к которому подключаются производители и потребители для отправки и получения сообщений. Множество брокеров образует кластер Kafka, что обеспечивает отказоустойчивость и масштабируемость.

5.Partition (Партиция):

1. Партиция в Kafka - это физический раздел топика, который содержит данные. Топики могут быть разделены на несколько партиций, что позволяет распределять нагрузку и увеличивать пропускную способность. Потребители могут подписываться на конкретные партиции в топике. Ключевая концепция в Kafka - это параллелизм, где данные разделяются между несколькими партициями.

Характеристики

1. Высокая пропускная способность и масштабируемость (горизонтальная)
2. Устойчивость и отказоустойчивость (хранение данных на диске и репликация)
3. Разделение данных на темы (топики)
4. Публикация-подписка
5. Задержки в реальном времени
6. Множество клиентов и языков программирования
7. Экосистема инструментов
8. Хранилище данных событий
9. Интеграция с Big Data и облачными решениями
10. Поддержка транзакций



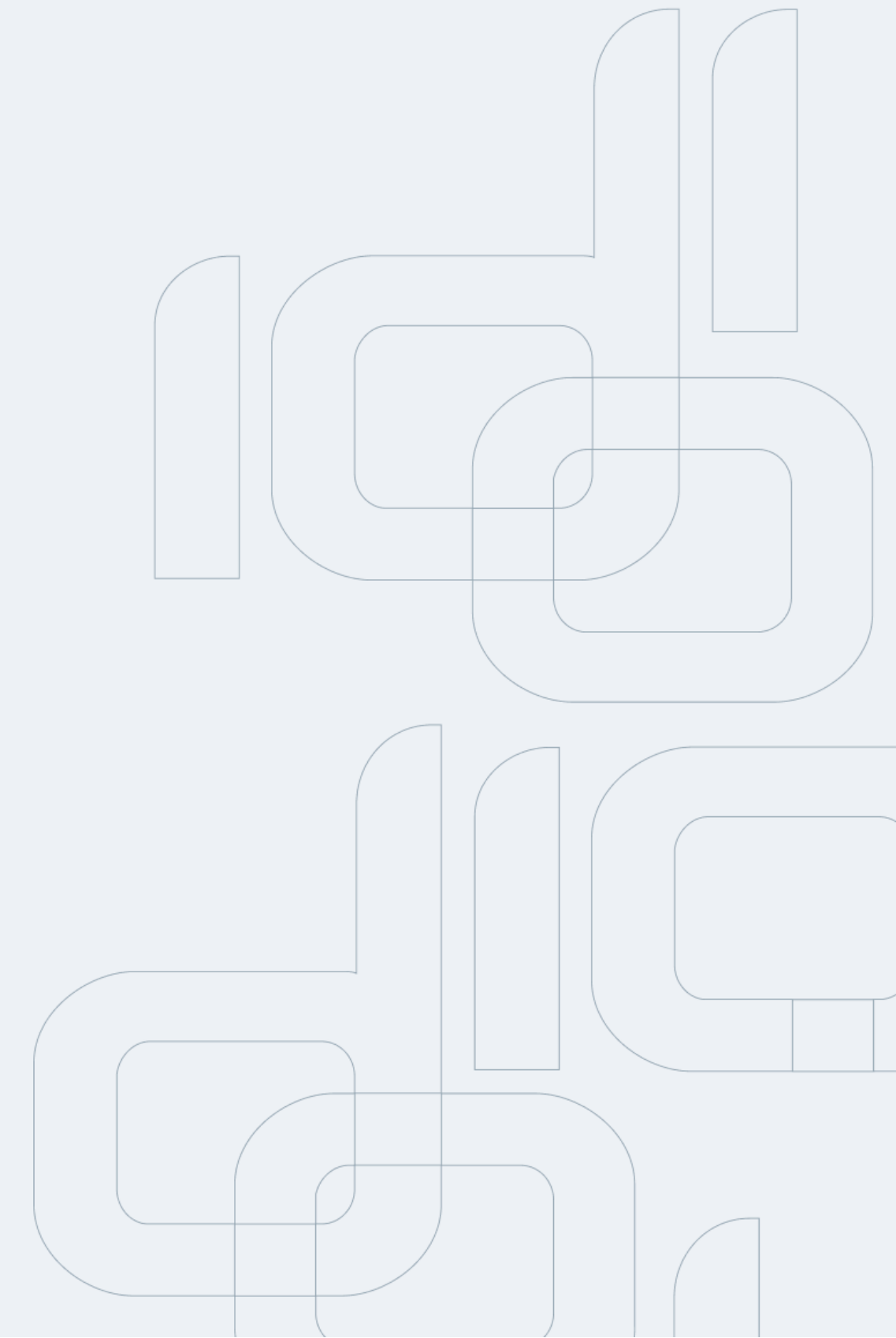
Сценарии использования

1. Поточковая обработка и анализ данных
2. Системы мониторинга и логирования
3. Агрегация данных из различных источников
4. Журнал событий и аудит
5. Системы реакции на события
6. Системы уведомлений и мгновенных сообщений
7. Системы микросервисов
8. Интеграция с облачными сервисами



Особенности применения

1. Разделение доступа
2. Группы клиентов
3. Вычитка старых сообщений
4. Вычитка одного сообщения несколькими клиентами
5. Гарантия порядка сообщений



Спасибо
за внимание

Мухамадуллин Ильяс
ilyasmukh@icloud.com