Брокеры сообщений в сервисной архитектуре

Андрей Бураков

TELEGRAM @and_burakov TG_CHANNEL @another_sa

Flow 2022

Более 10 лет в роли разработчика, аналитика, тимлида

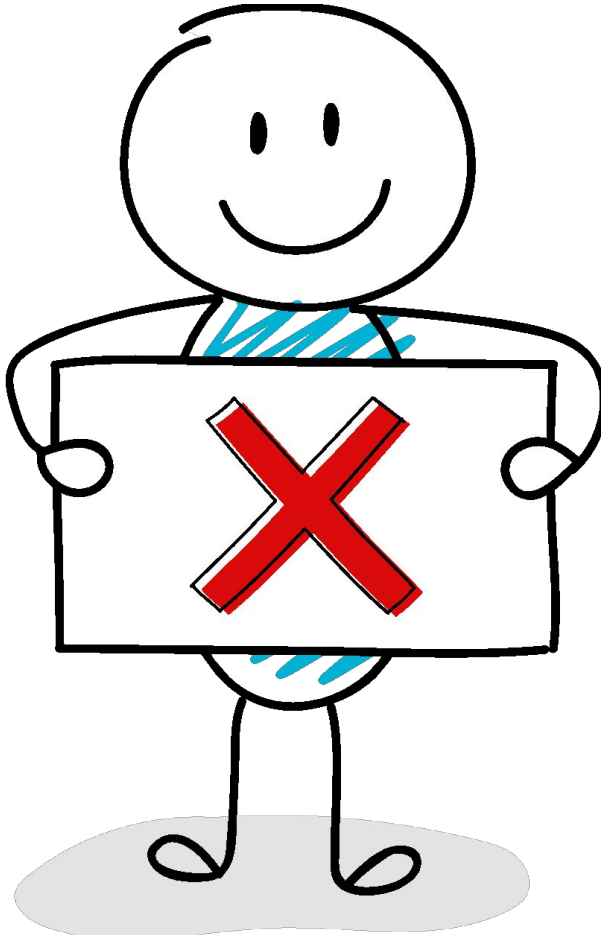
- ex-VK Pay, создал отдел системного и бизнес анализа
- ex-Buyk, tech product owner
- Основатель школы анализа и проектирования Next**Way**
- Автор канал Yet Another SA - @another_sa

Зачем мы здесь?

Что будет

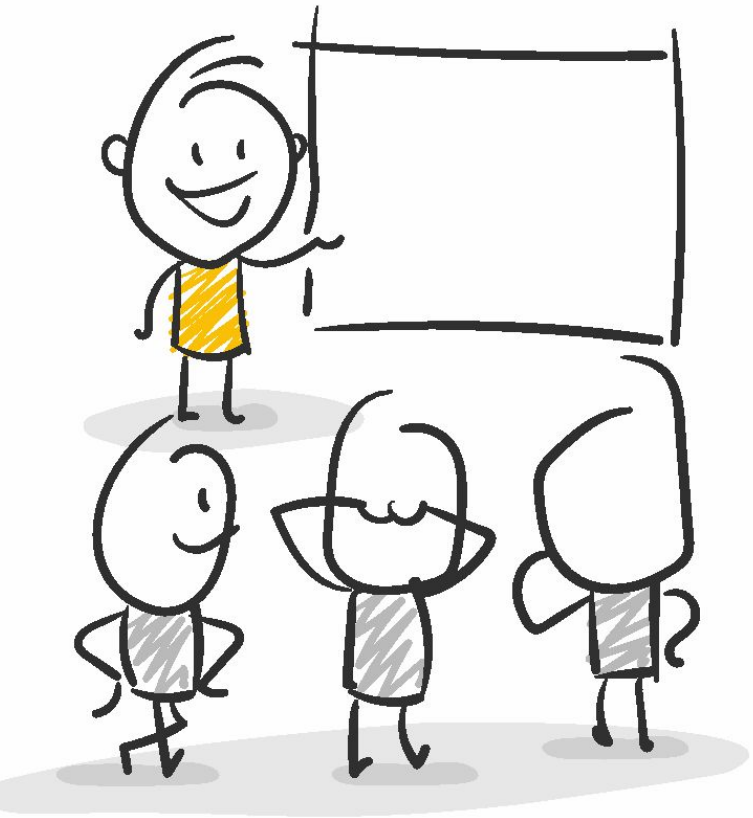
- Базовые паттерны использования брокеров
- Взаимодействие брокера и сервисов
- Гарантии доставки, и на что это влияет

Чего не будет



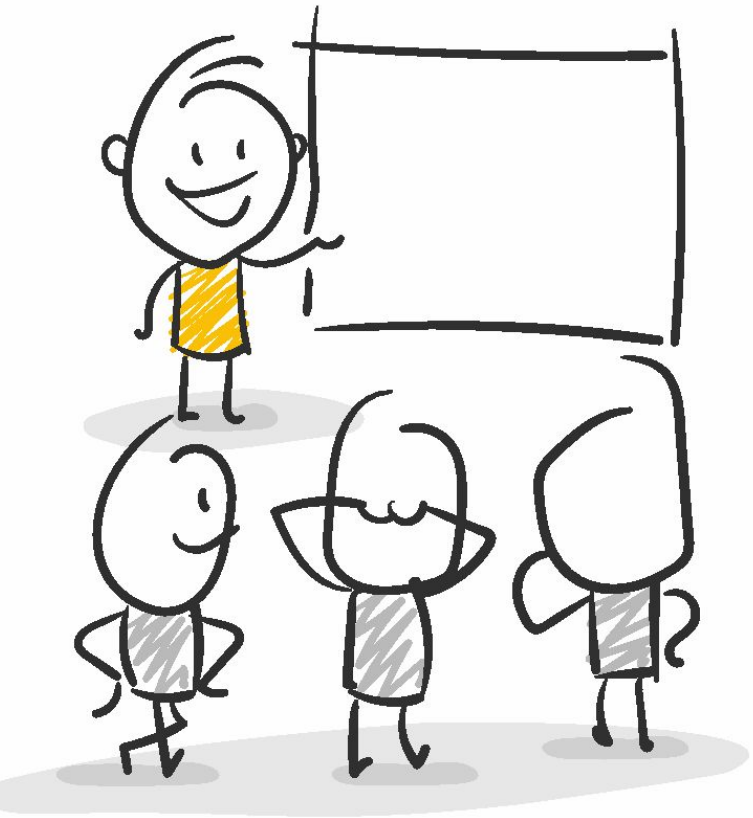
- Какой брокер лучше
- Как правильно использовать Kafka, RabbitMQ, etc.

Как работаем



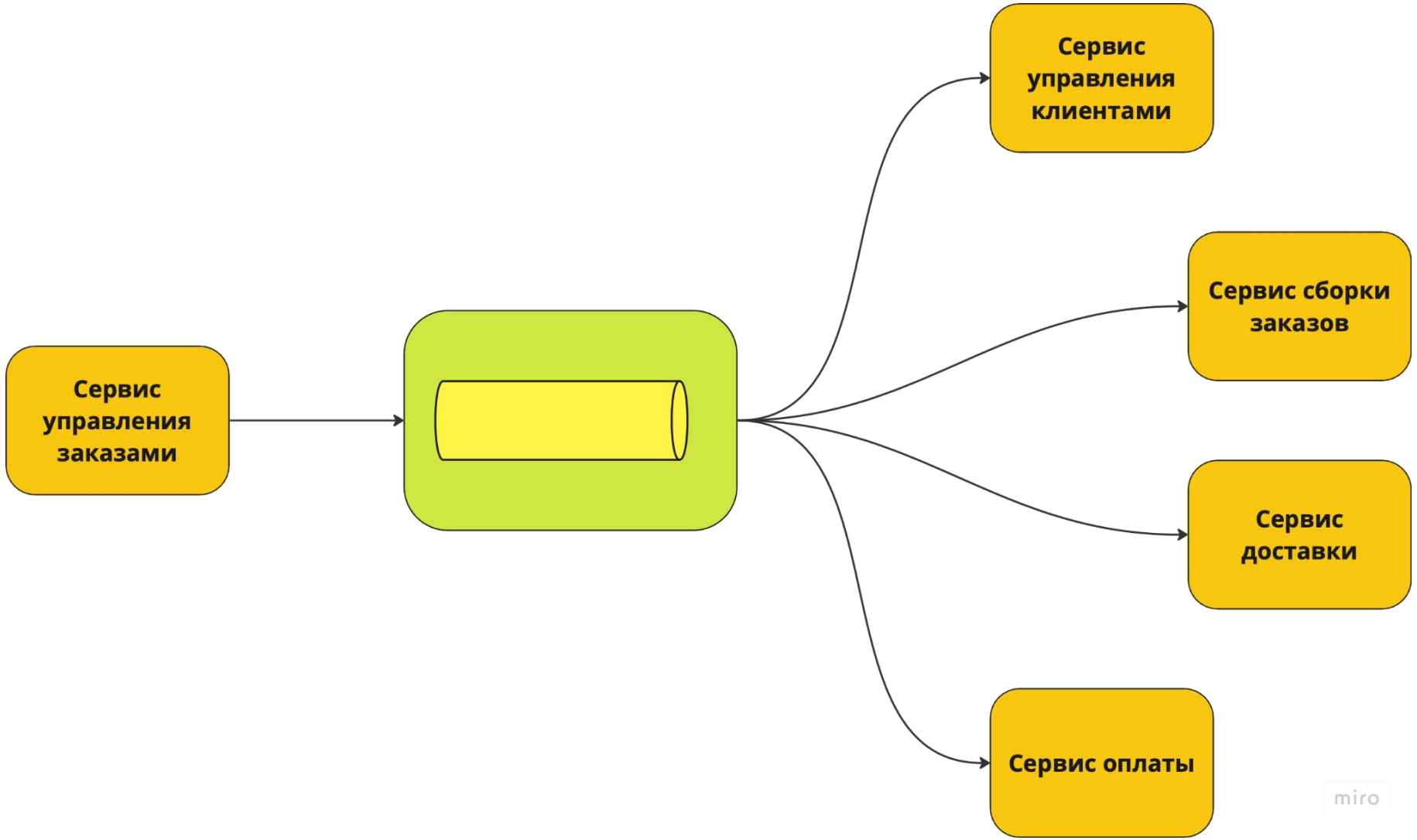
- Обсуждаем теорию
- Проектируем решение для кейса в команде
- Разбираем вместе результаты

Познакомимся?

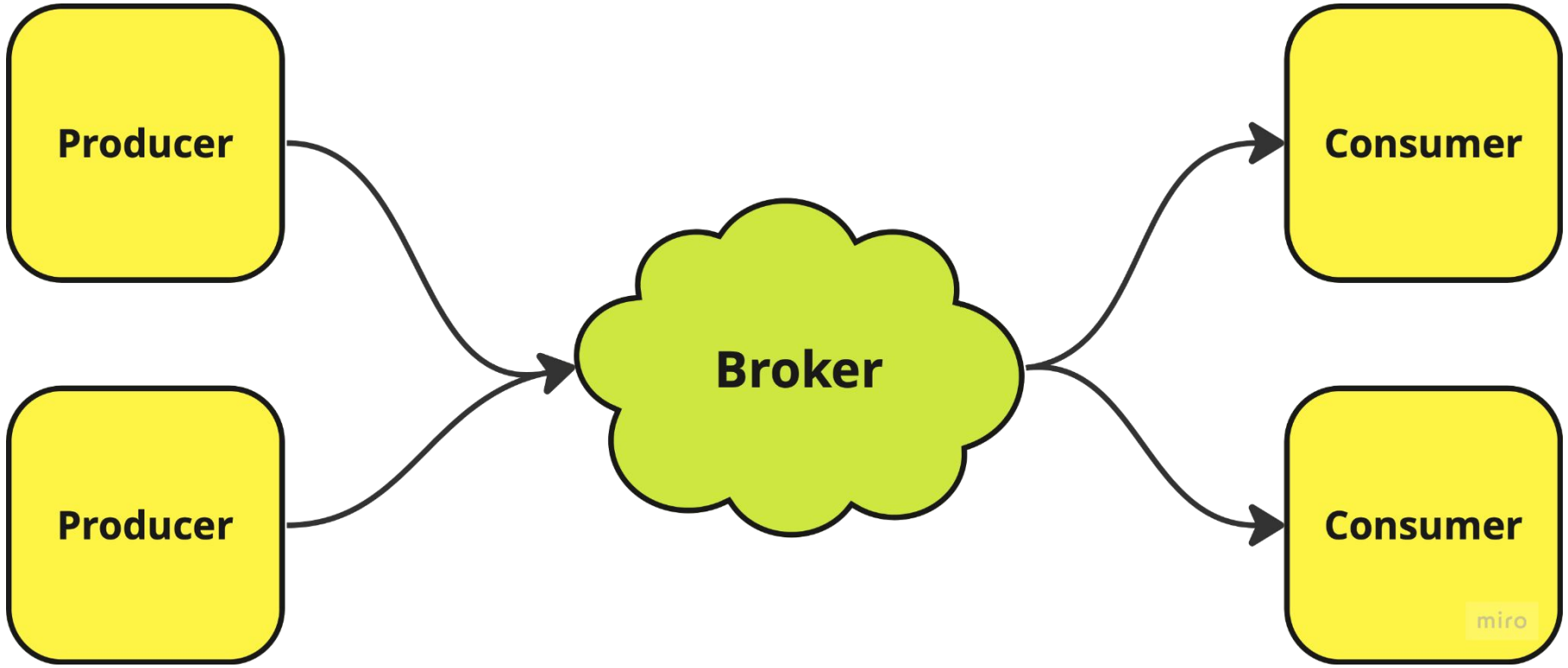


- Как меня зовут
- Над каким проектом или продуктом работаю
- Зачем мне

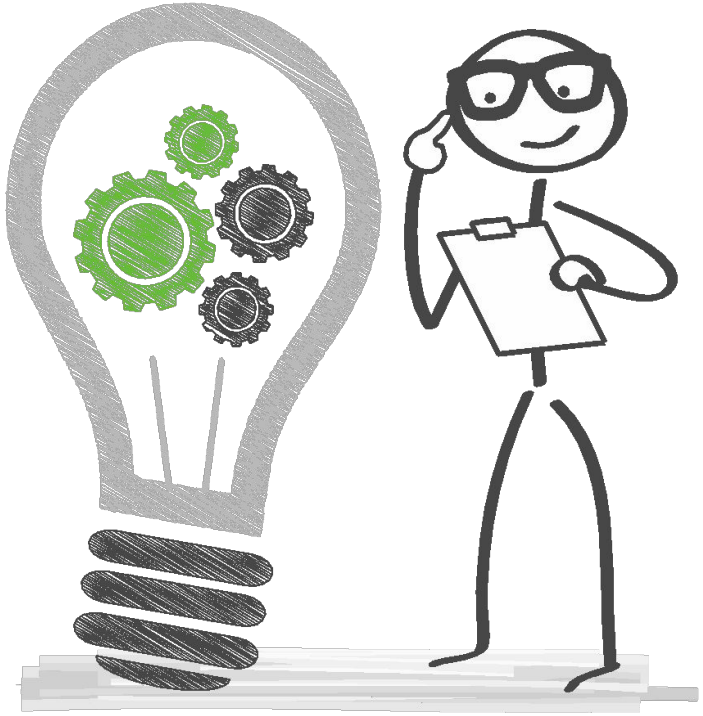
Доставка пиццы



Терминология

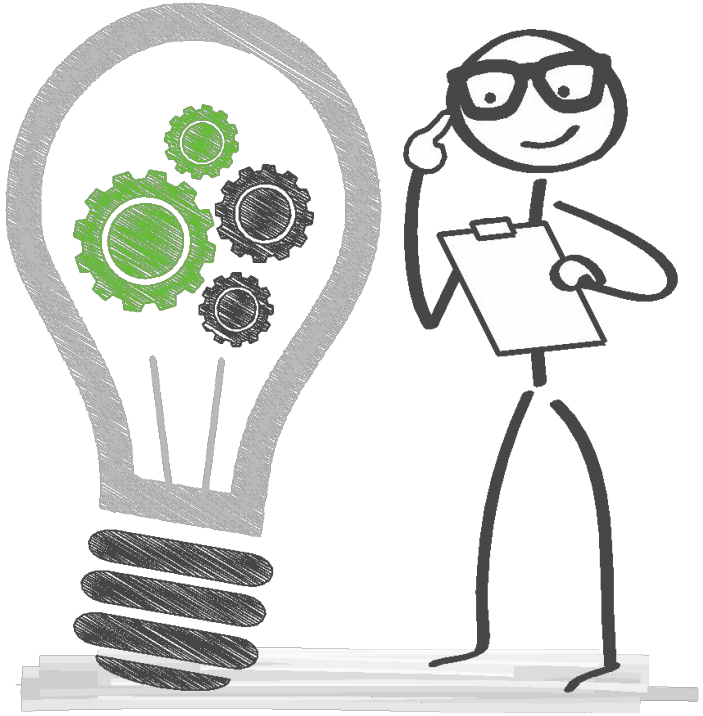


Базовые паттерны



- One Way
- Request - Reply
- Pub-Sub

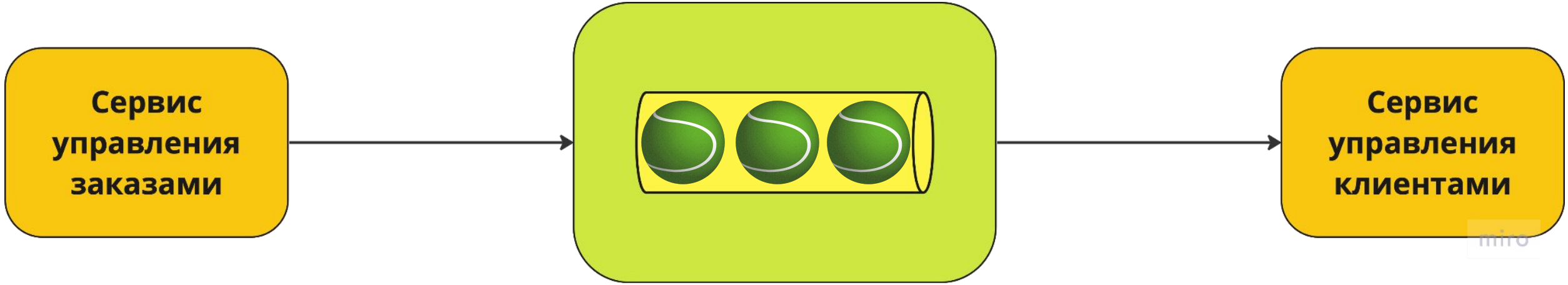
Очередь



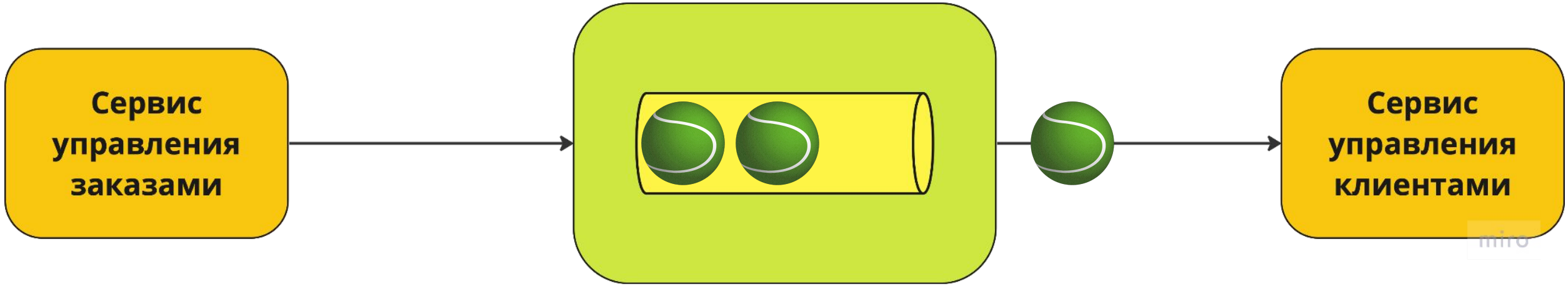
Очередь - паттерн, при котором брокер реализует First-In-First-Out логику передачи сообщений, где каждое сообщение передается одному потребителю.

При успешном чтении сообщение удаляется из очереди.

Очередь

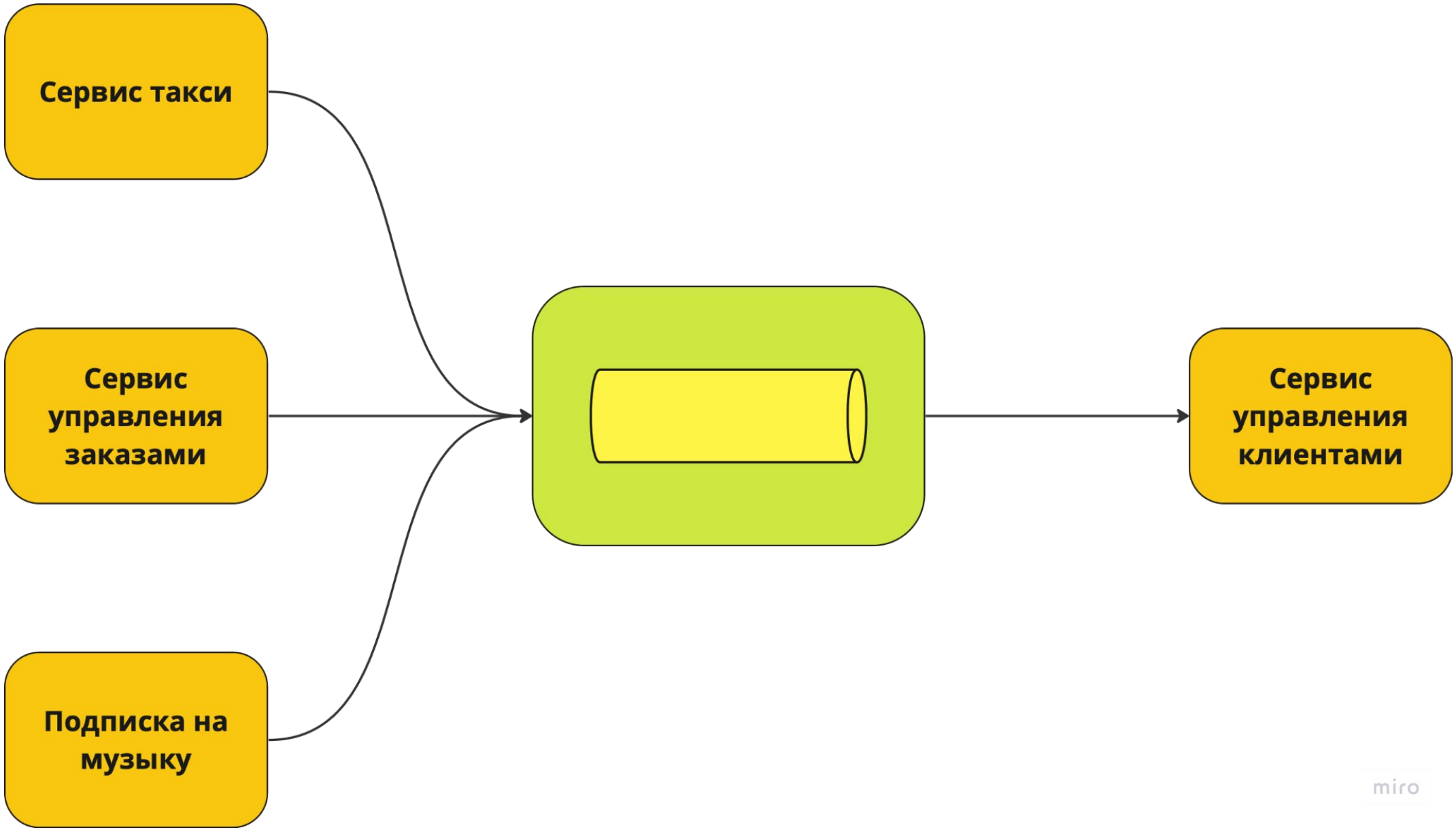

Сервис управления заказами
Сервис управления клиентами
miro

Очередь


Сервис управления заказами
Сервис управления клиентами
miro

Зачем это?

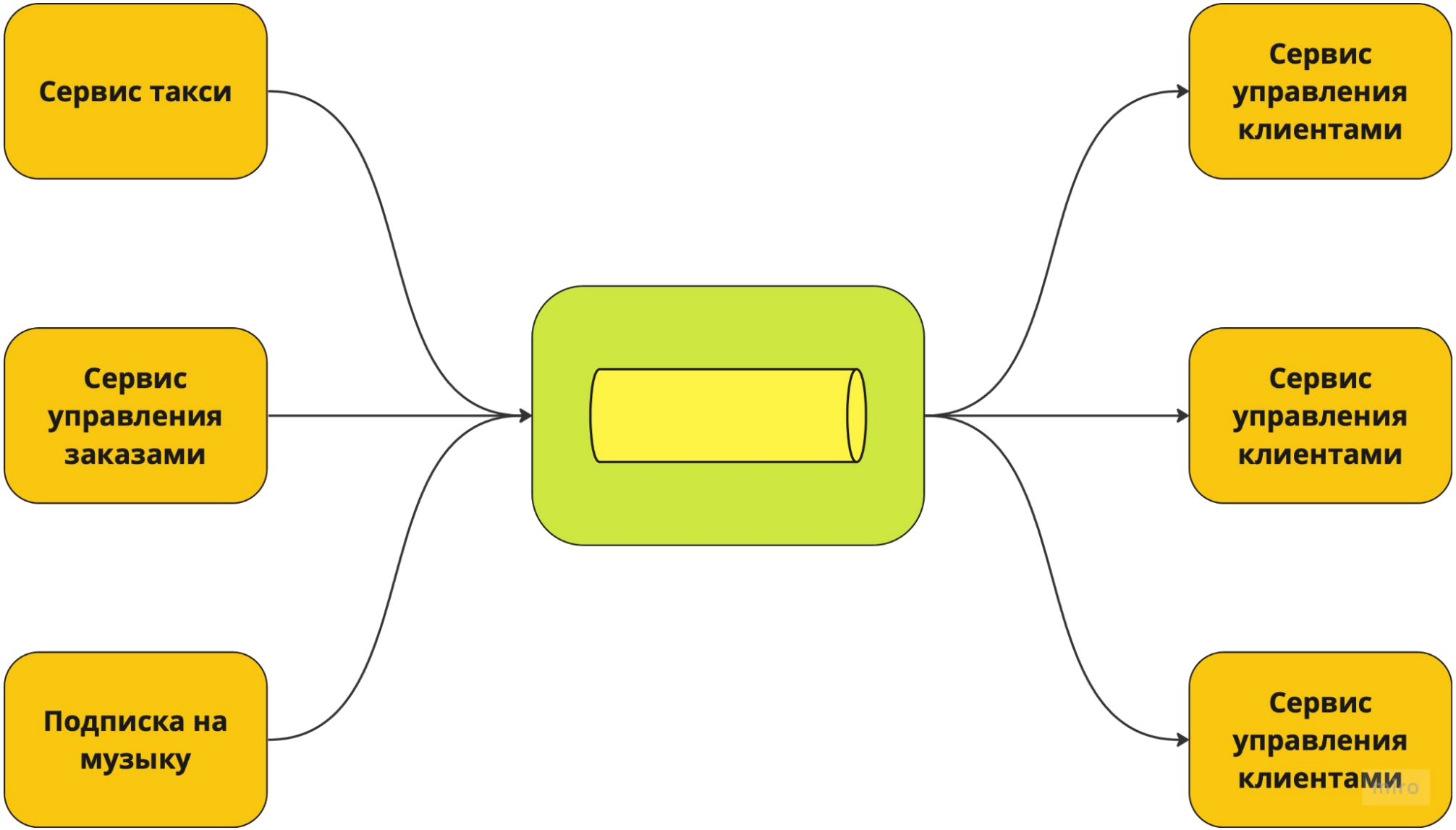
- Снимаем с клиента логику доставки сообщений
- Снижаем связанность между сервисами


Сервис такси
Сервис управления заказами
Подписка на музыку
Сервис управления клиентами

miro

Зачем это?

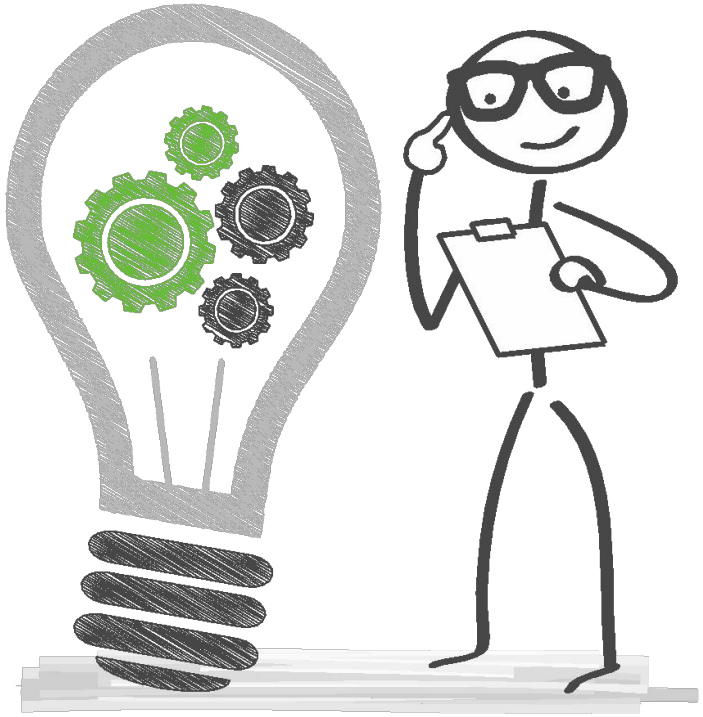
- Снимаем с клиента логику доставки запросов
- Снижаем связанность между сервисами
- Управление нагрузкой


Сервис такси
Сервис управления заказами
Подписка на музыку
Сервис управления клиентами
Сервис управления клиентами
Сервис управления клиентами

Зачем это?

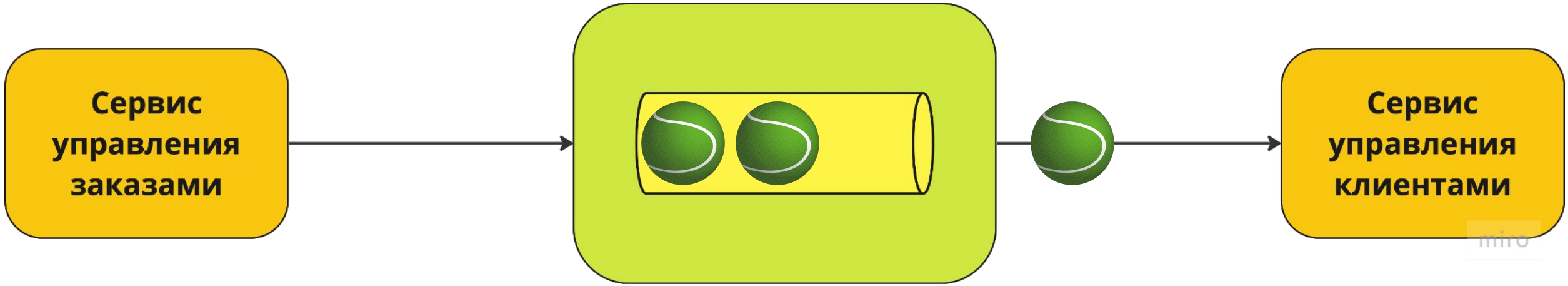
- Снимаем с клиента логику доставки запросов
- Снижаем связанность между сервисами
- Управление нагрузкой
- Упрощаем масштабирование

Использование очереди

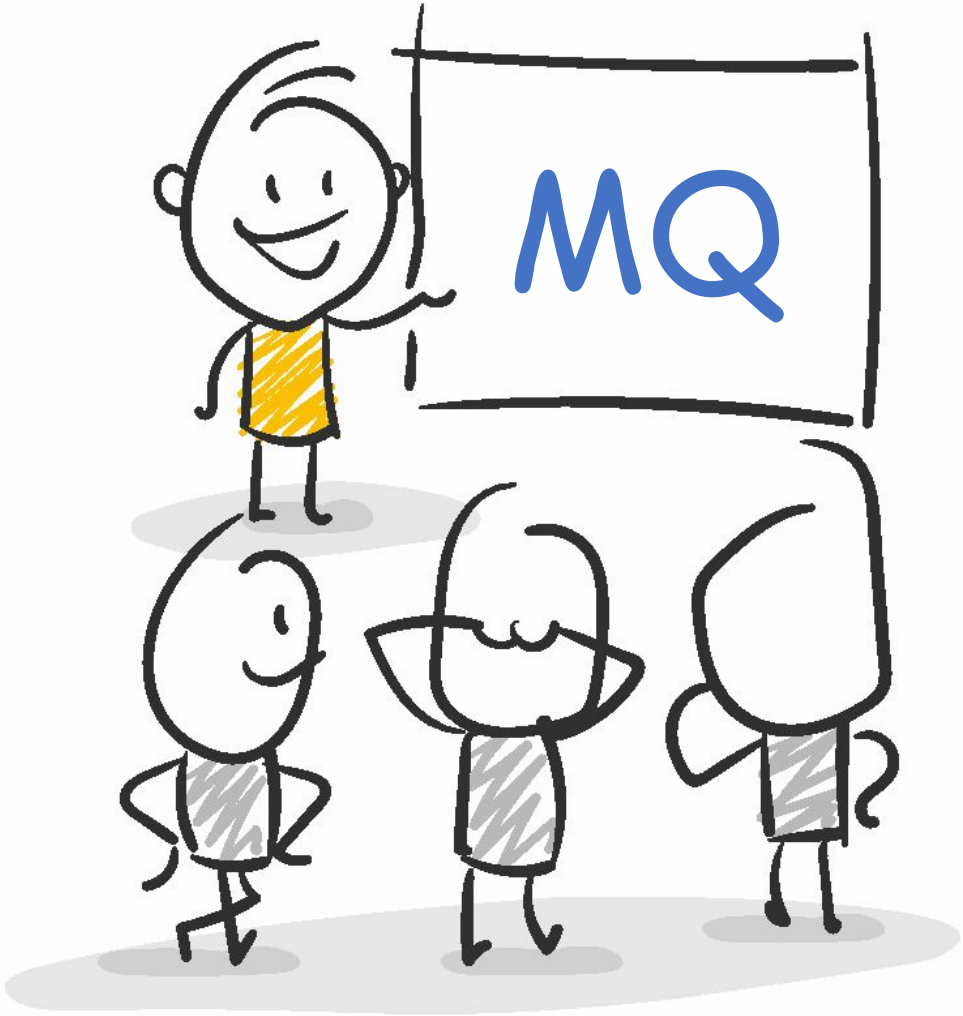


- **One Way -** поставщик отправляет сообщение и не ждет реакции на него
- **Request-Reply** - поставщик отправляет сообщение-запрос и ждет ответа

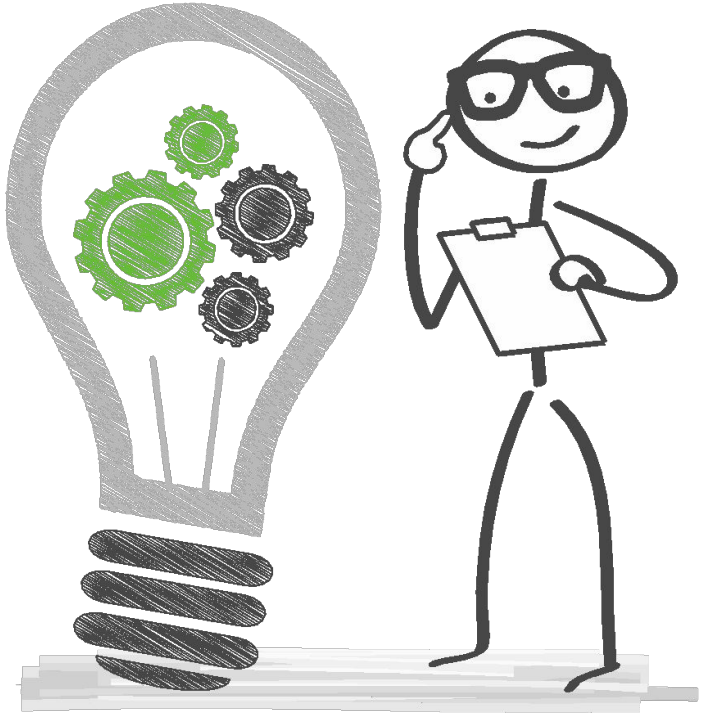
One Way


Сервис управления заказами
Сервис управления клиентами
miro

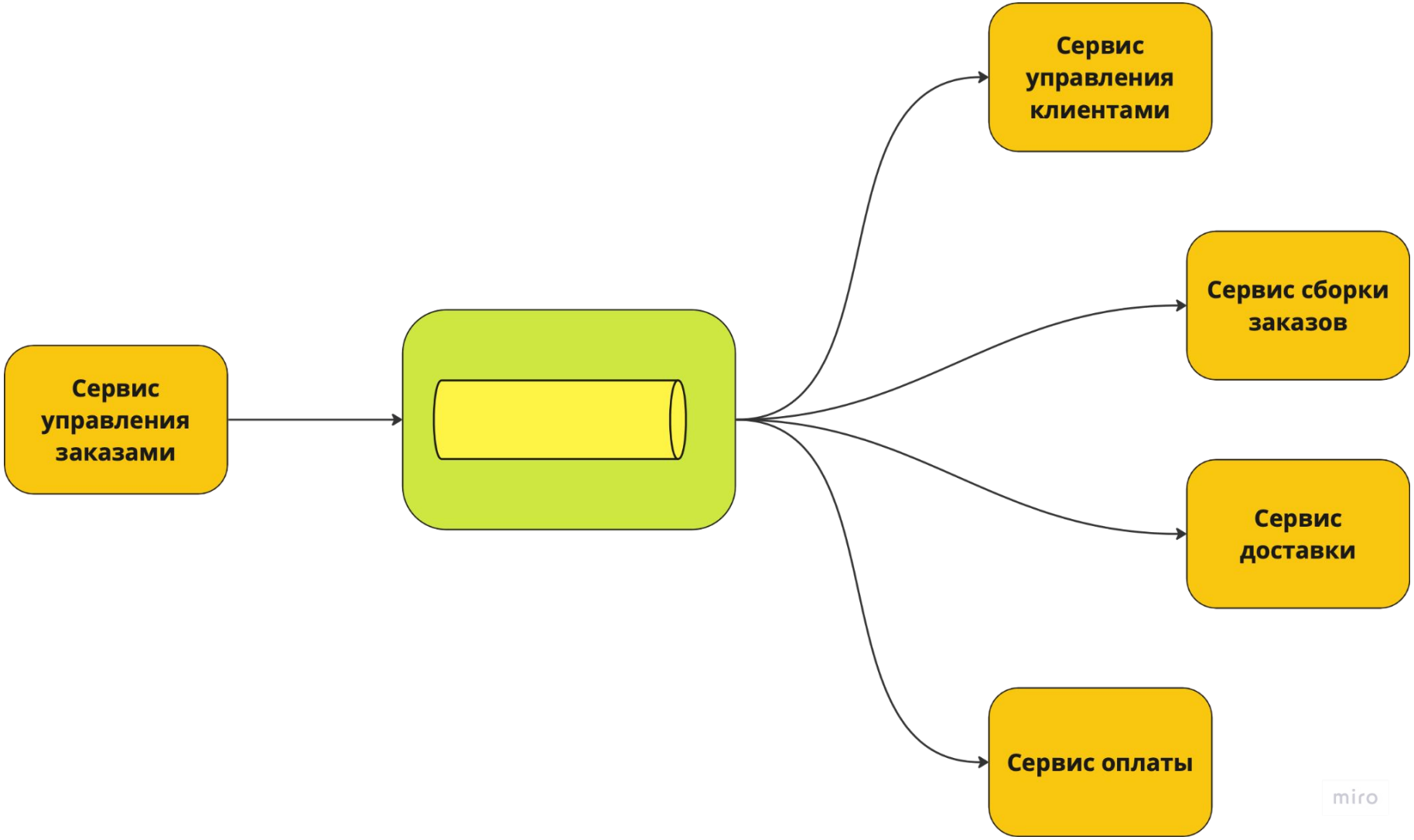
Практика 1



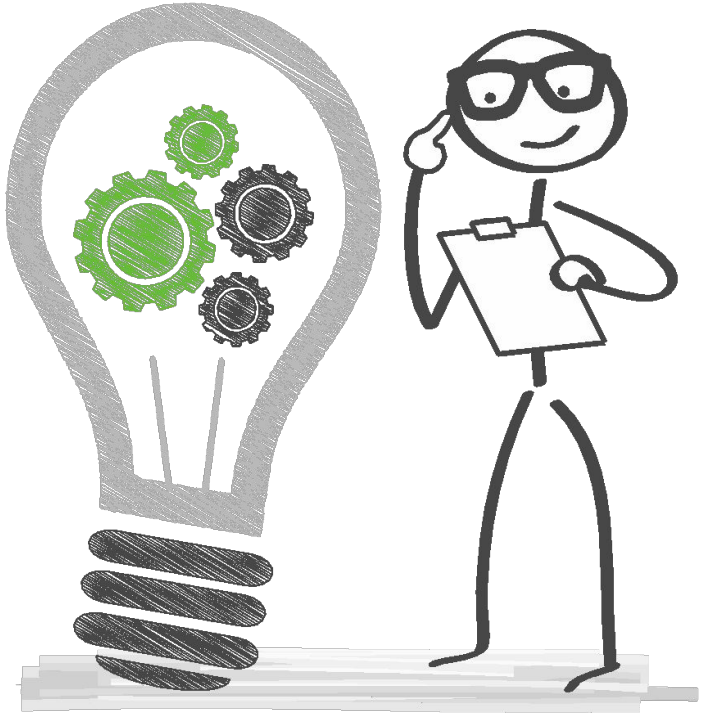
О чем нужно помнить



- Механизмы обработки проблемных и невалидных сообщений
- Таймауты и механизмы повторов


Сервис управления заказами
Сервис управления клиентами
Сервис сборки заказов
Сервис доставки
Сервис оплаты
miro

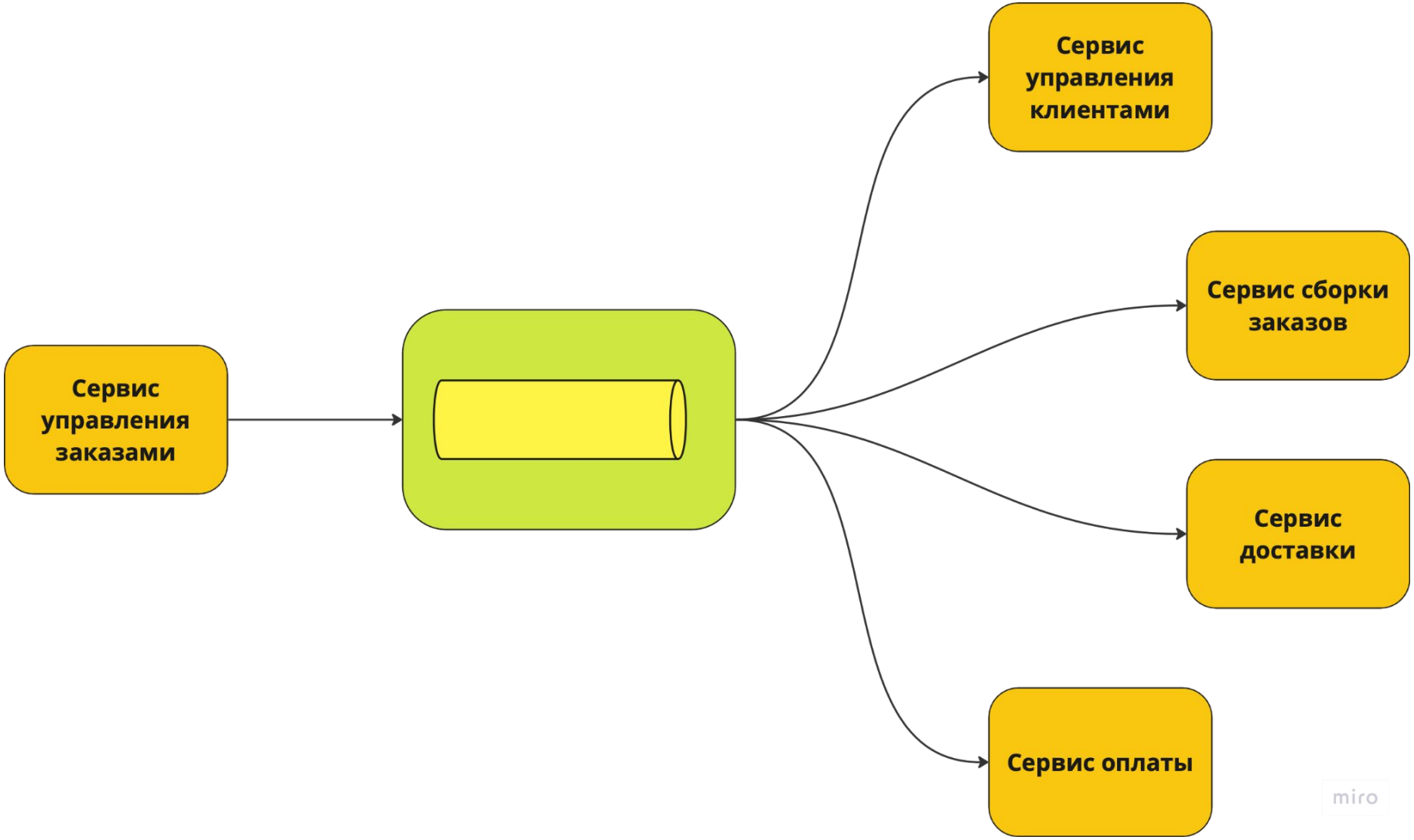
Подписка



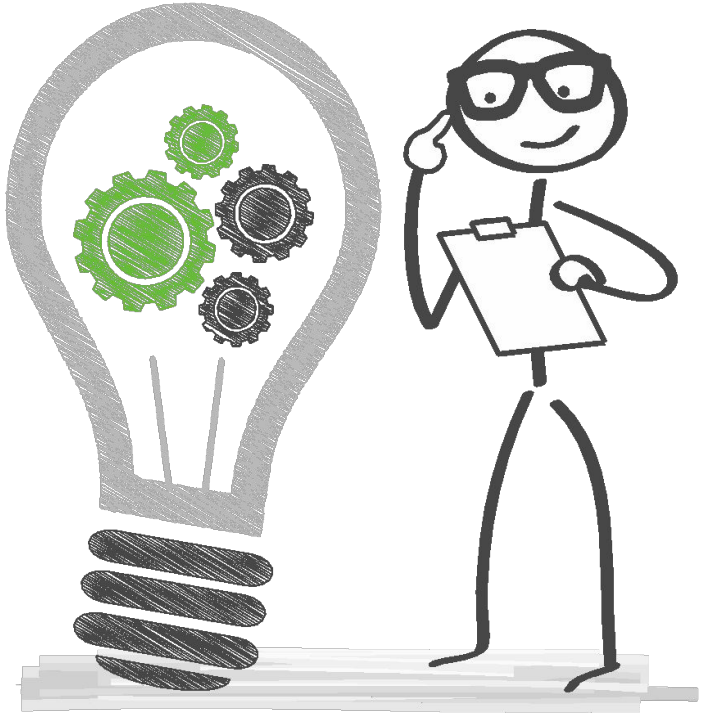
Подписка - паттерн, при котором брокер реализует логику передачи одного сообщения сразу нескольким потребителям.

Зачем это?

- Снимаем с клиента логику доставки запросов
- Снижаем связанность между сервисами
- Реализуем логику массовой отправки сообщений

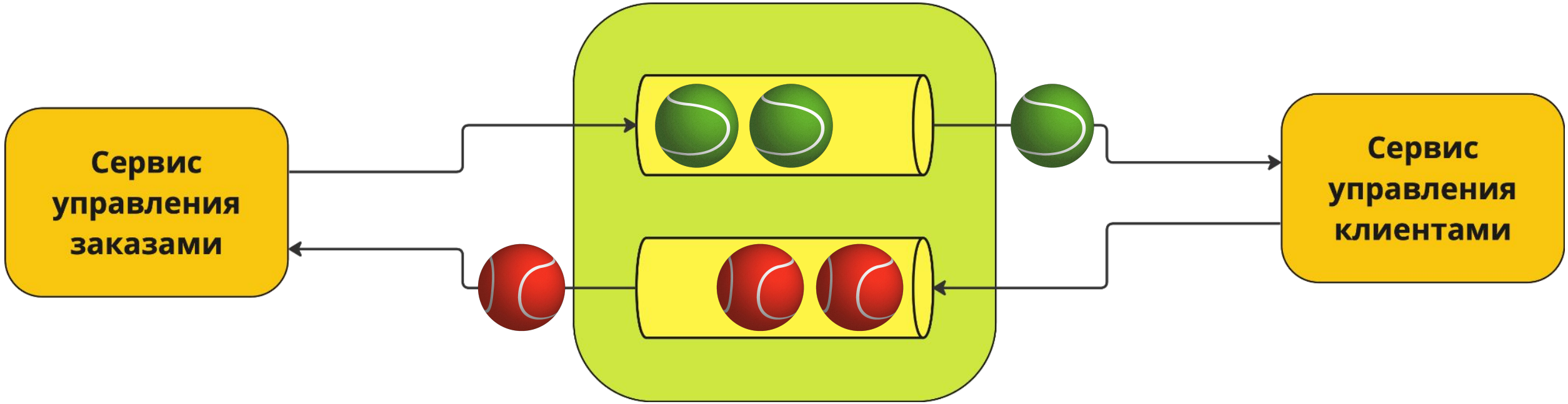

Сервис управления заказами
Сервис управления клиентами
Сервис сборки заказов
Сервис доставки
Сервис оплаты
miro

О чем нужно помнить

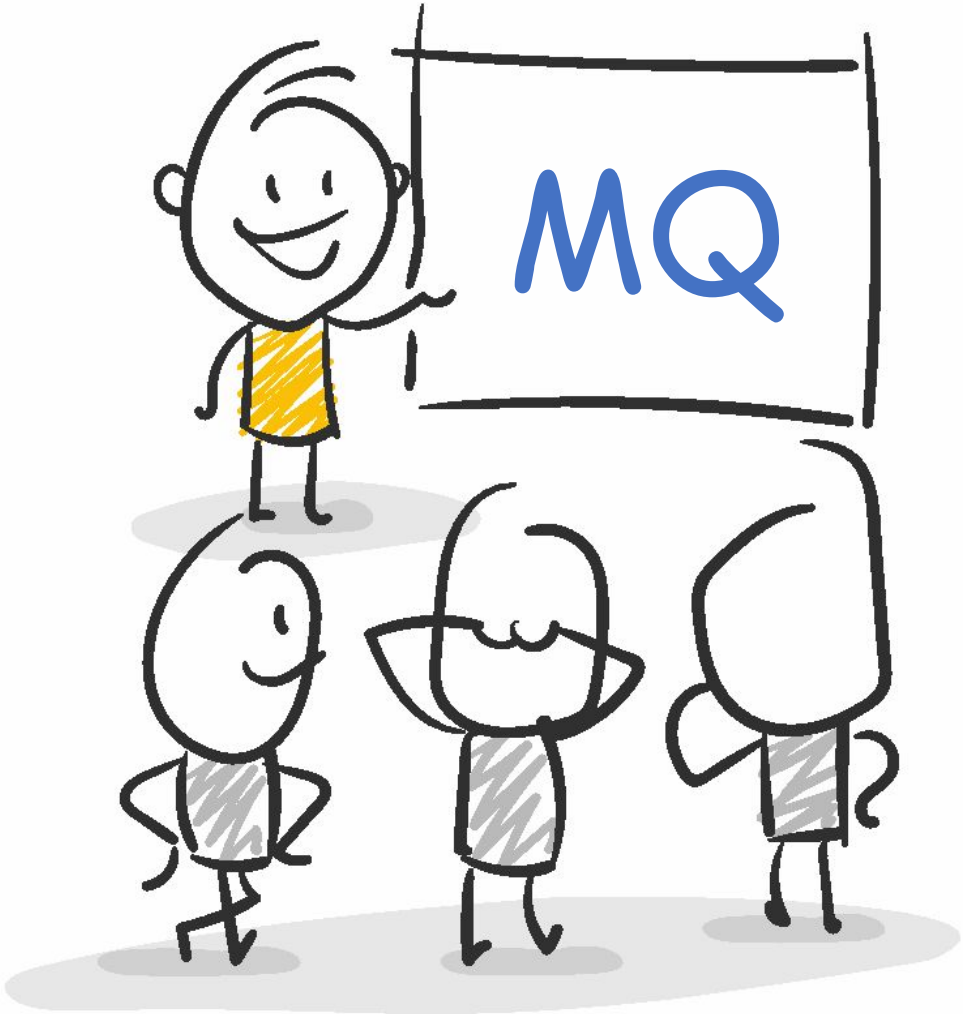


- Механизмы обработки проблемных и невалидных сообщений
- Таймауты и механизмы повторов

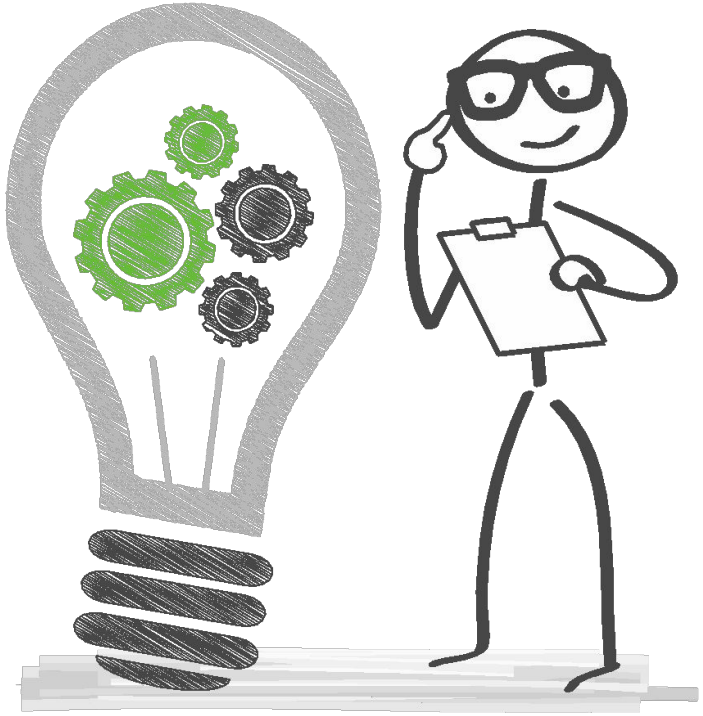
Request-Reply



Практика 2

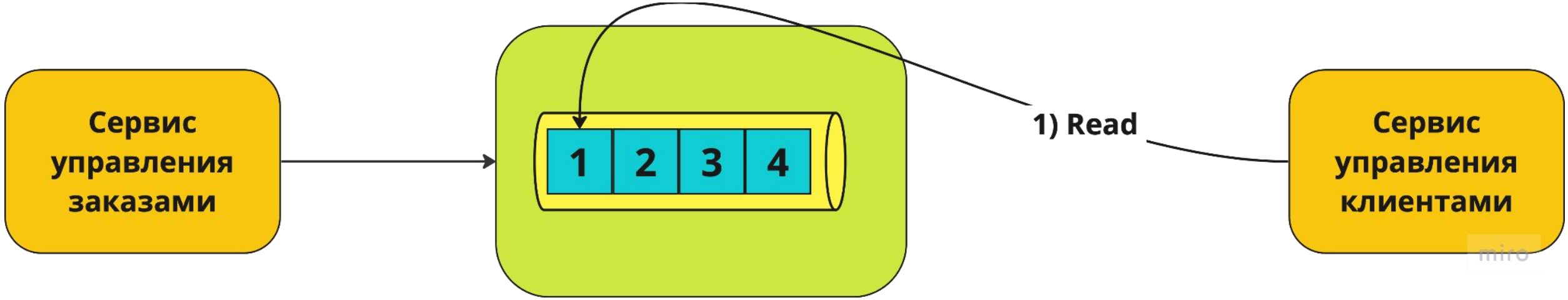


О чем нужно помнить

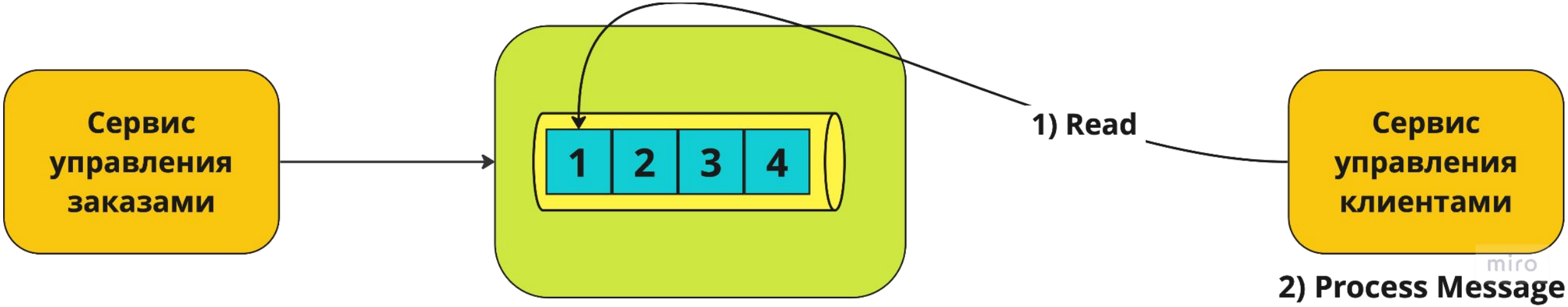


- Логика разделения каналов получателей
- Сопоставление запроса и ответа

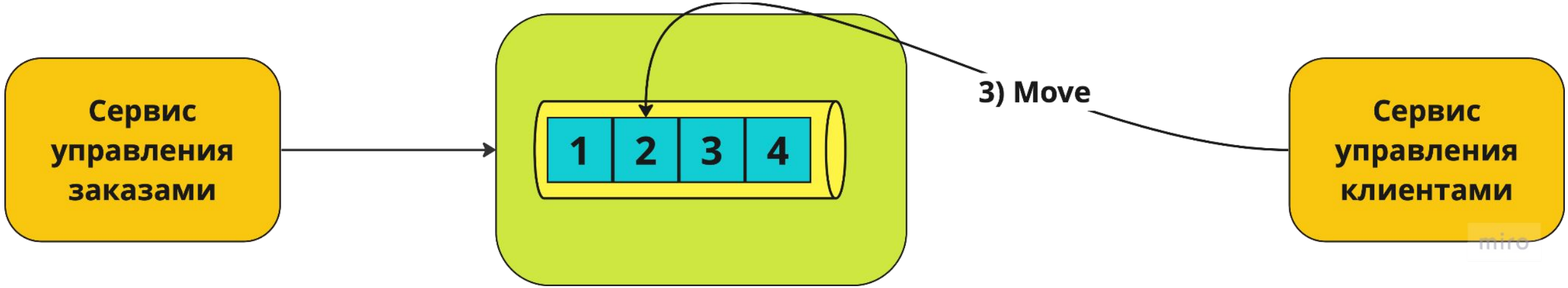
Топики Kafka-style

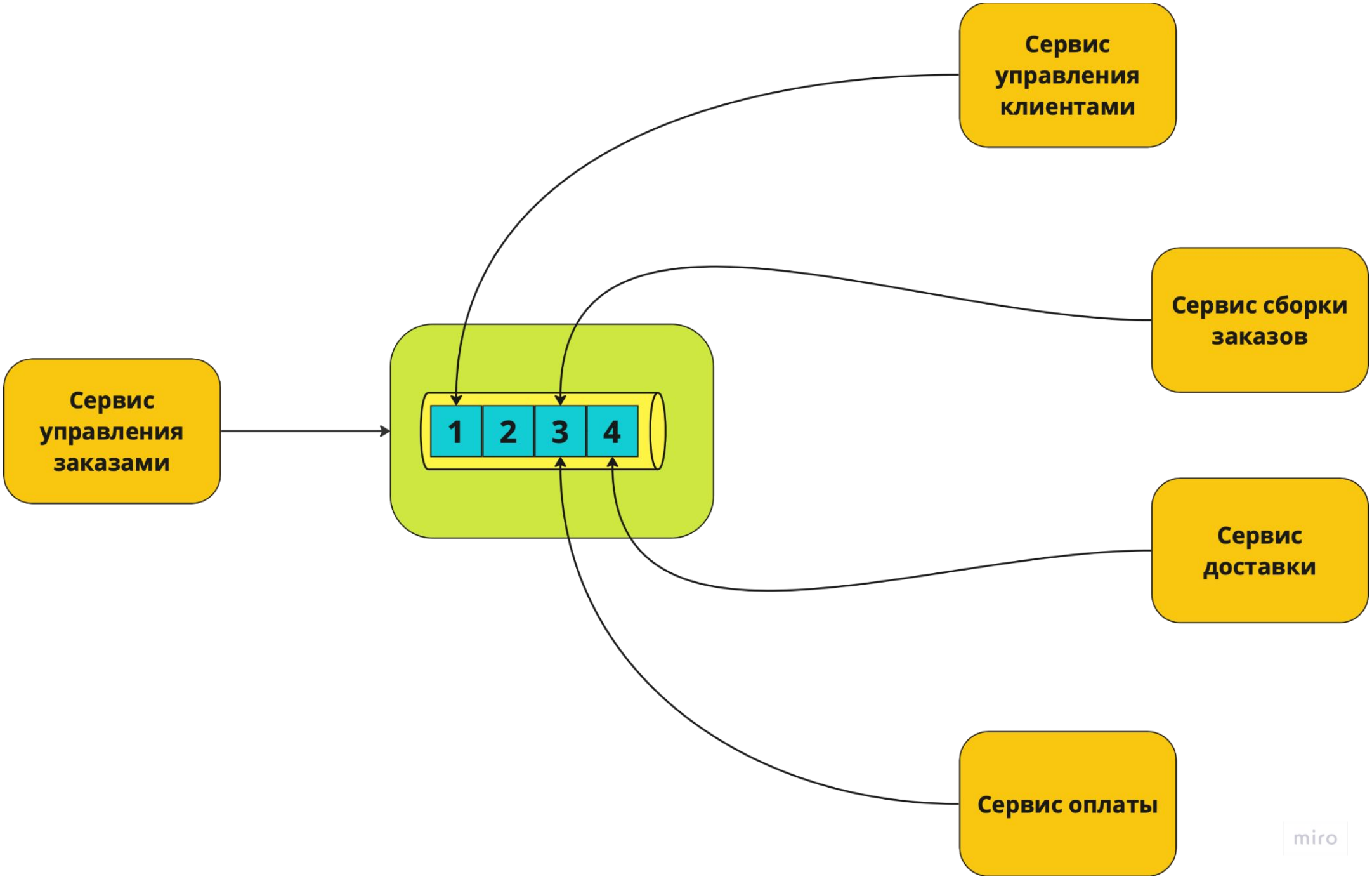


Топики Kafka-style

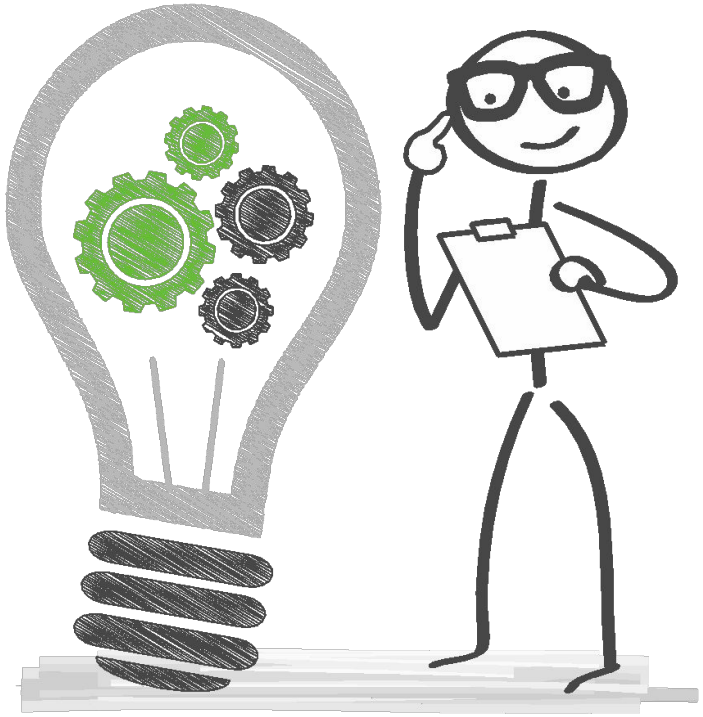


Топики Kafka-style




Сервис управления клиентами
Сервис сборки заказов
Сервис управления заказами
1
2
3
4
Сервис доставки
Сервис оплаты
miro

Модель потребления



Модель потребления сообщений - это первое решение, которое нам нужно принять.

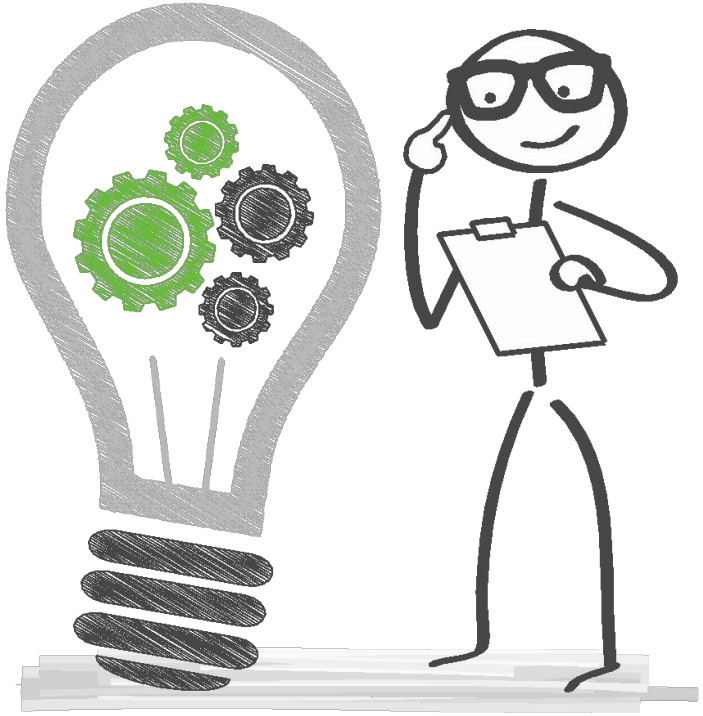
- Queue
  - One Way
  - Request - Reply
- Pub-Sub
  - Хранение сообщений
  - Удаление сообщений

Как сообщение попадает в брокер?

Как сообщение попадает к получателю?

Что может пойти не так?

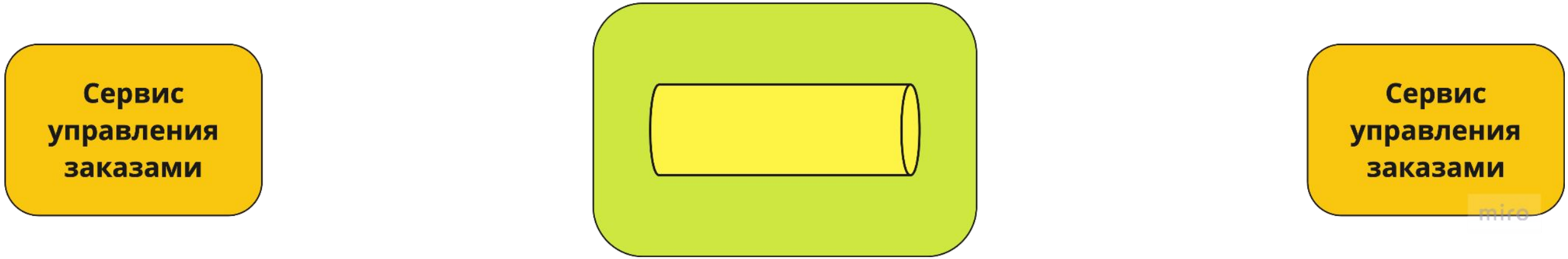
Взаимодействие с брокером



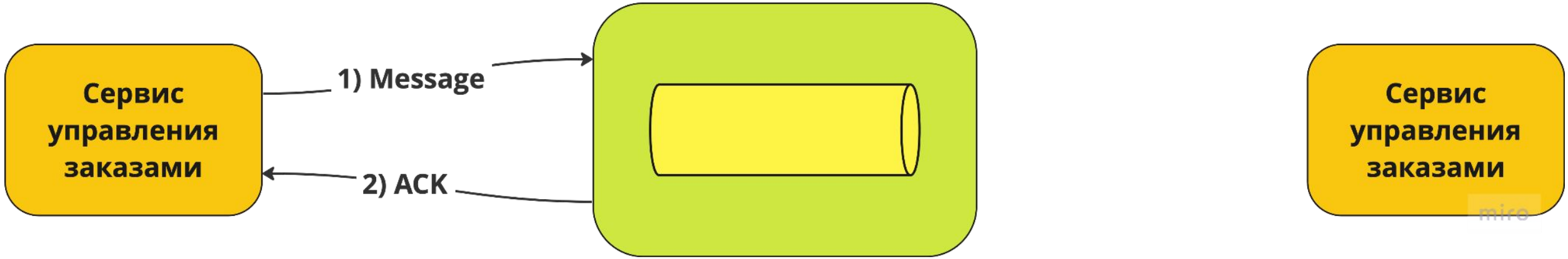
Потребители и поставщики взаимодействуют с брокером через сетевые протоколы.

Точно так же, как и сервисы

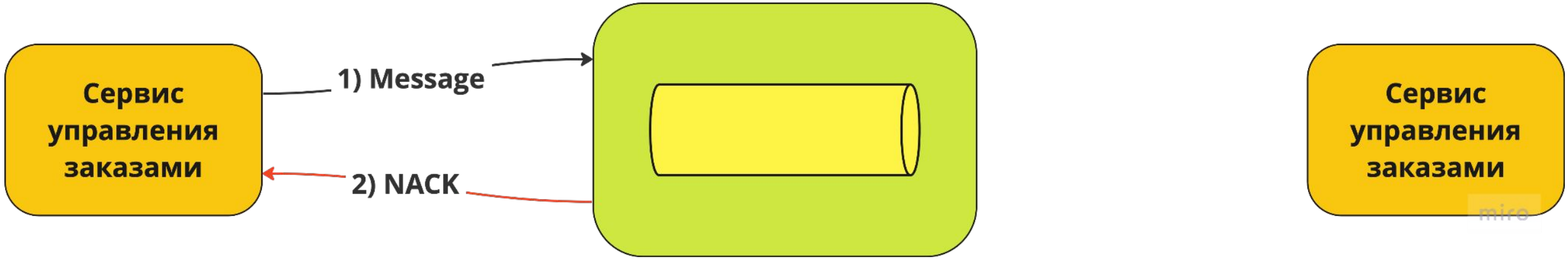
Отправка в брокер



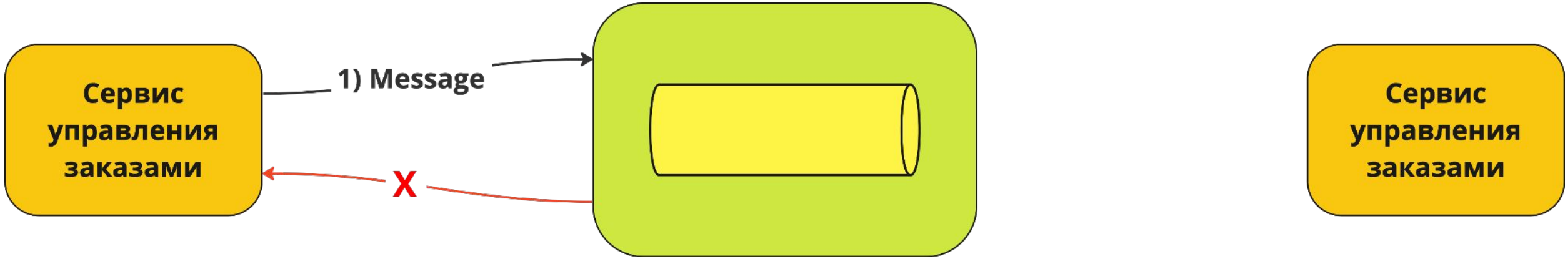
Отправка в брокер



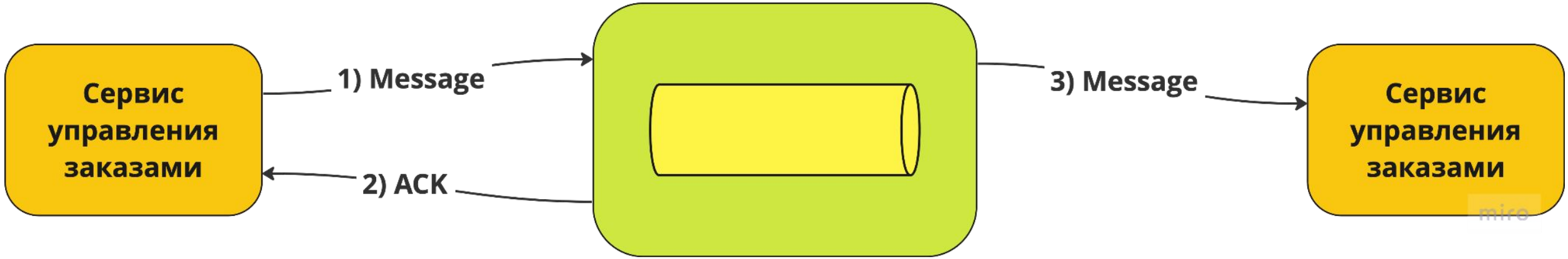
Отправка в брокер



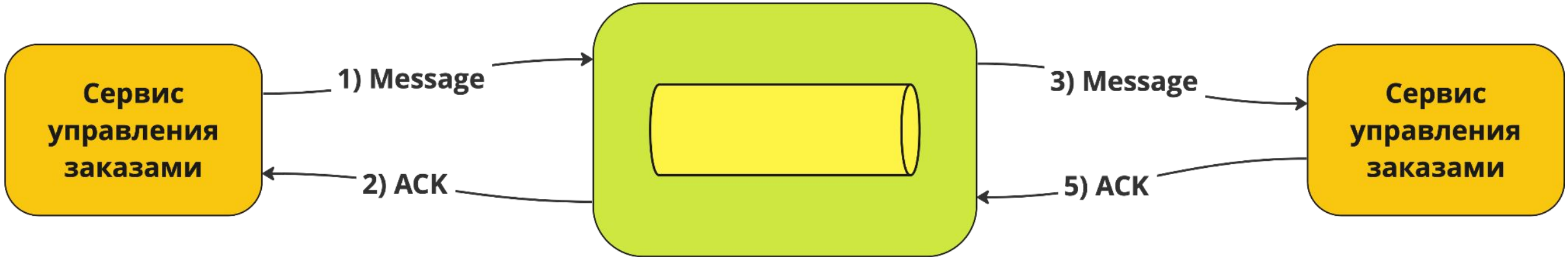
Отправка в брокер



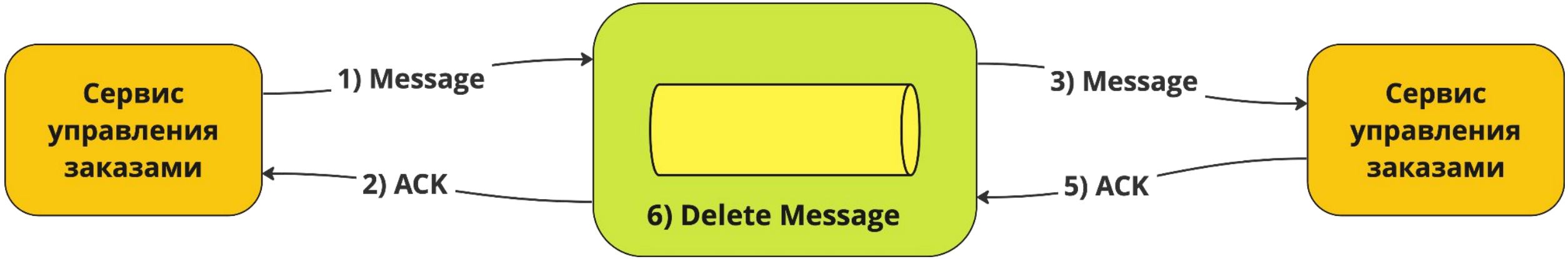
Получение из брокера



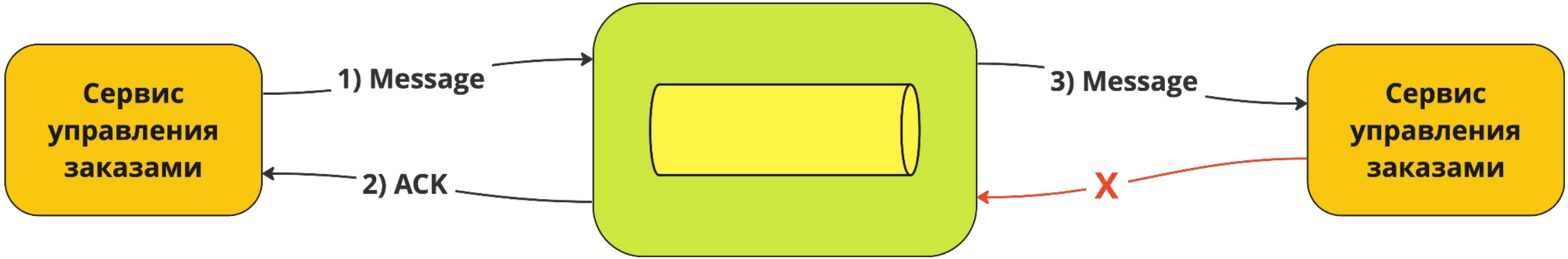
Получение из брокера



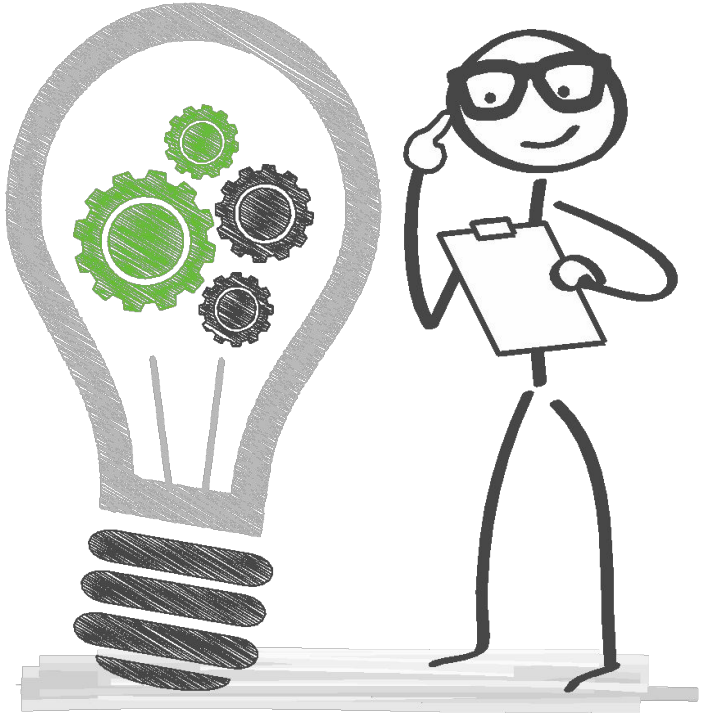
Получение из брокера



Получение из брокера

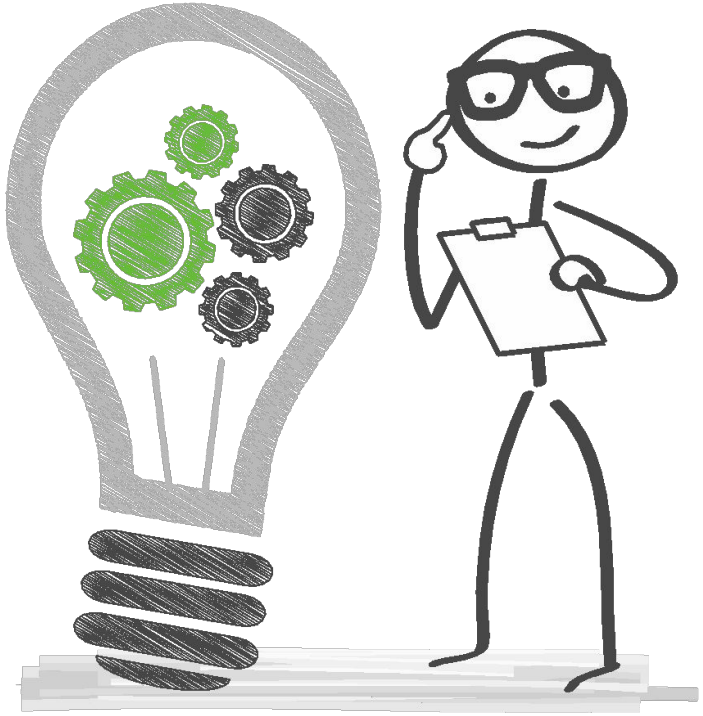


At least once



At least once - режим работы, при котором гарантируется, что сообщение будет доставлено <u>минимум</u> один раз.

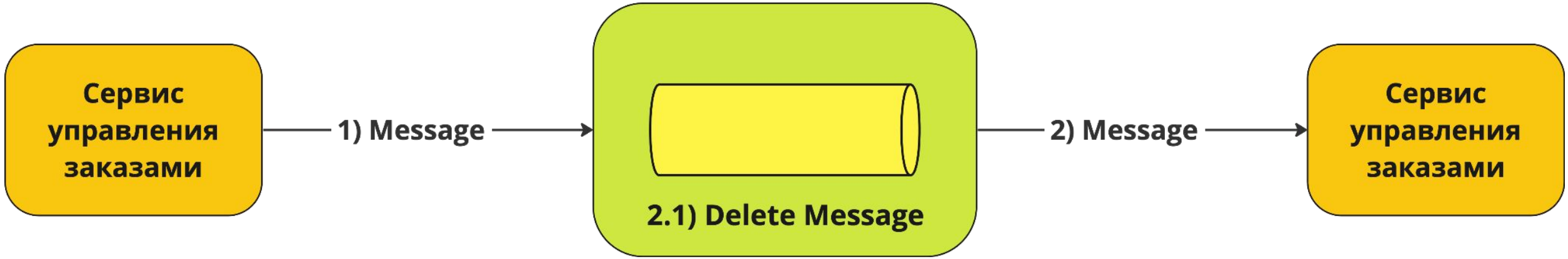
At least once



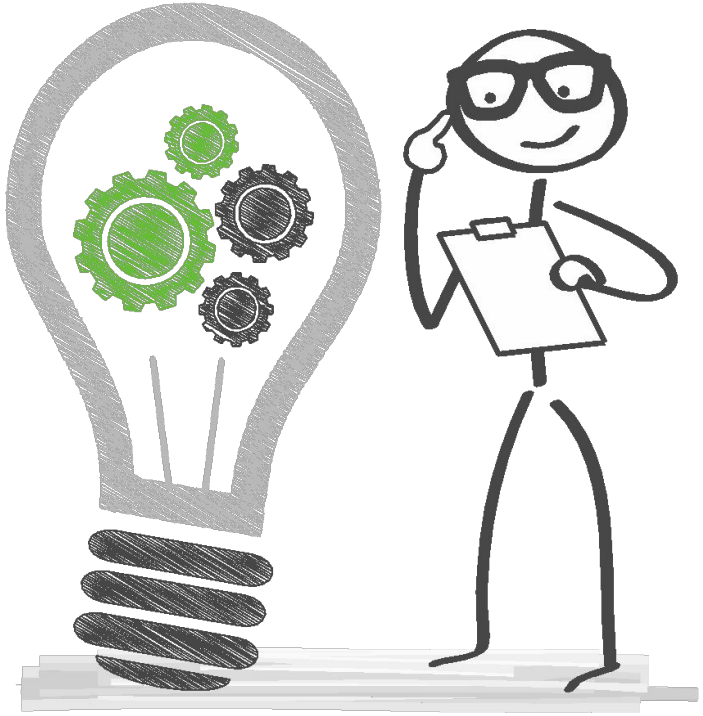
At least once - режим работы, при котором гарантируется, что сообщение будет доставлено <u>минимум</u> один раз.

При этом одно и то же сообщение может быть доставлено несколько раз одному или разным потребителям.

Отправка и получение



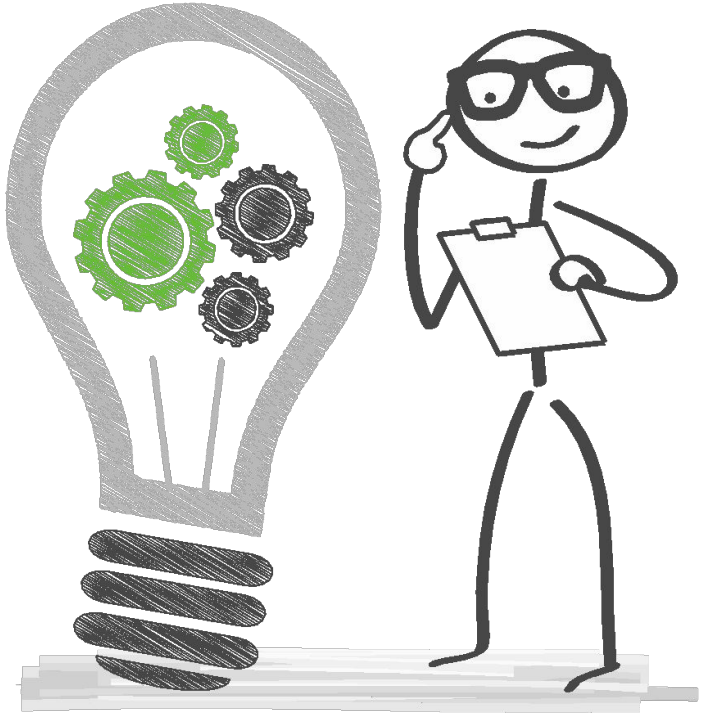
At most once



At most once - режим работы, при котором гарантируется, что сообщение будет доставлено <u>максимум</u> один раз.

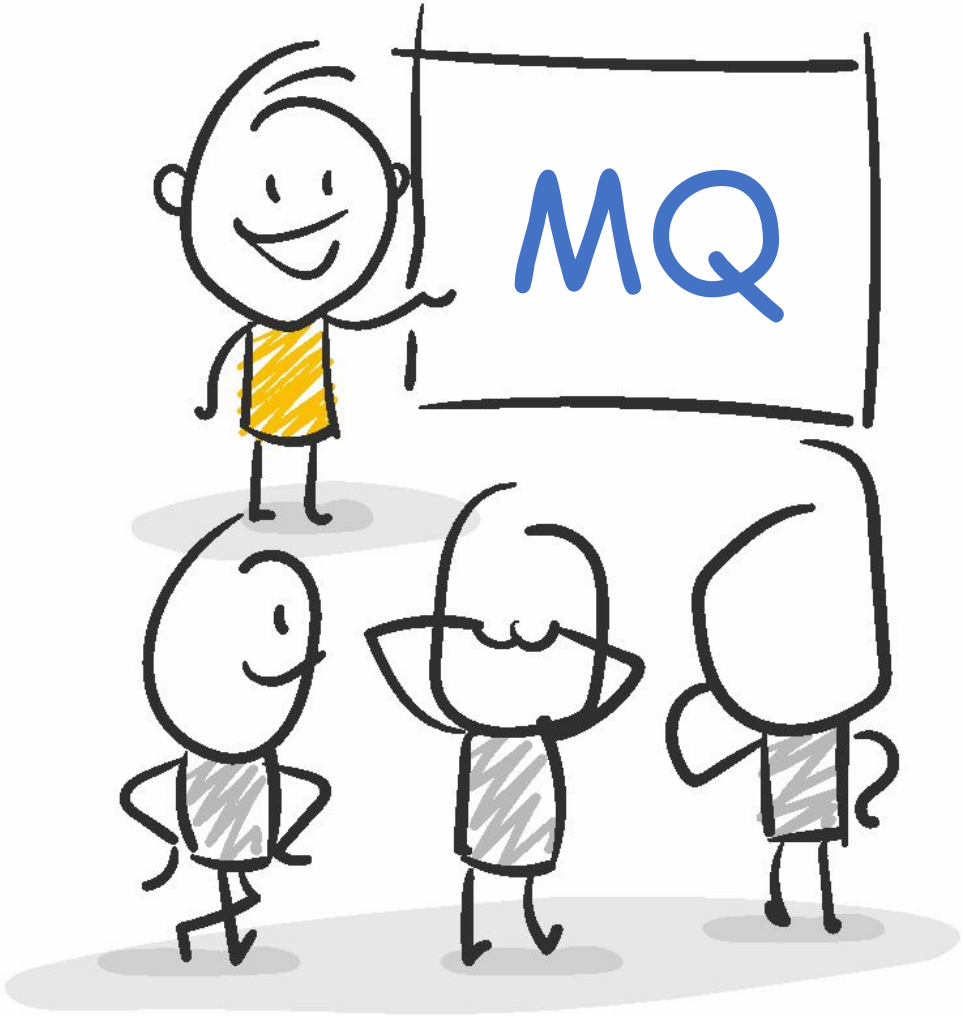
При этом сообщение может “потеряться”, и его не обработает ни один сервис.

Гарантии доставки

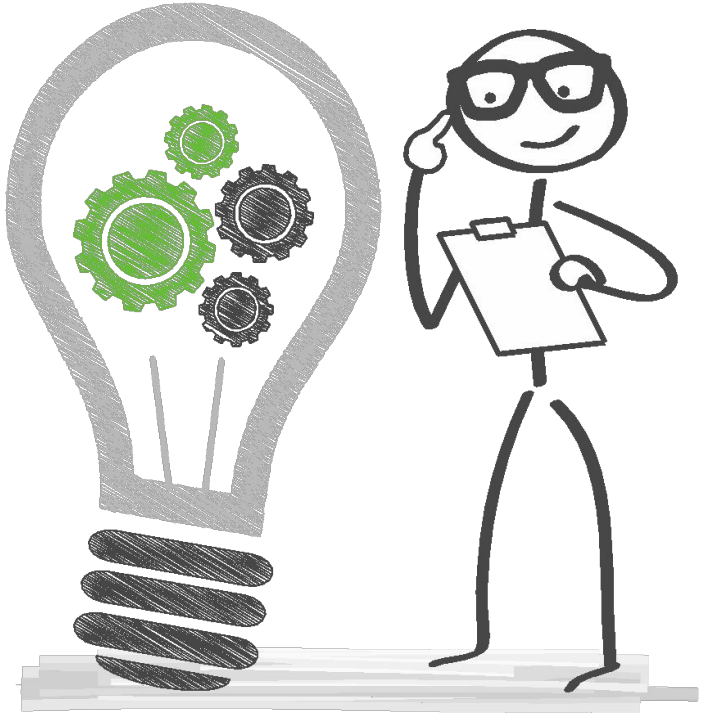


- at least once
- at most once
- exactly once

Практика 3

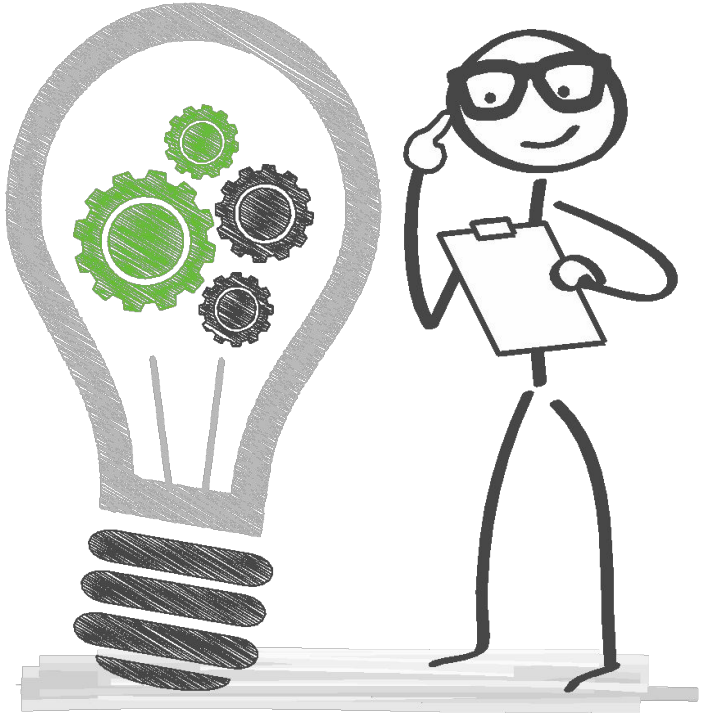


At least once



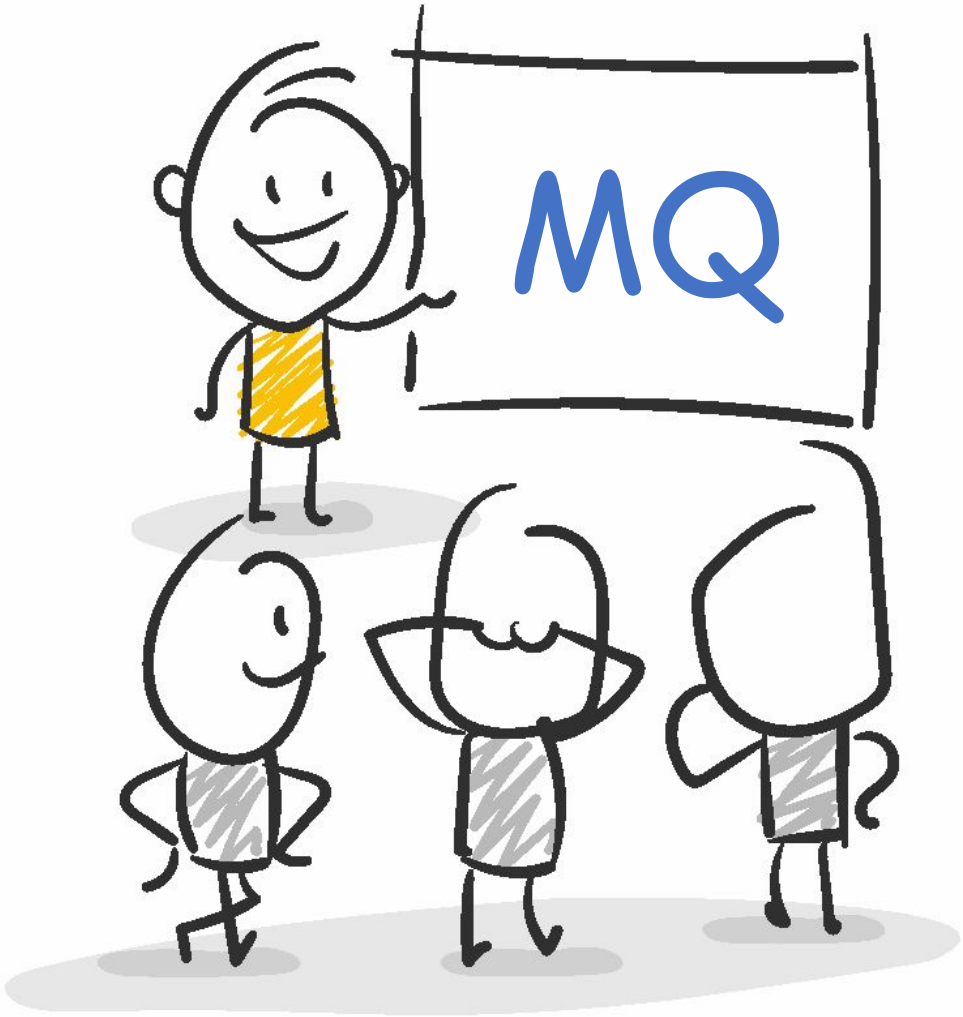
При работе с **at least once** доставкой важно обеспечить идемпотентность сервисов.

At most once

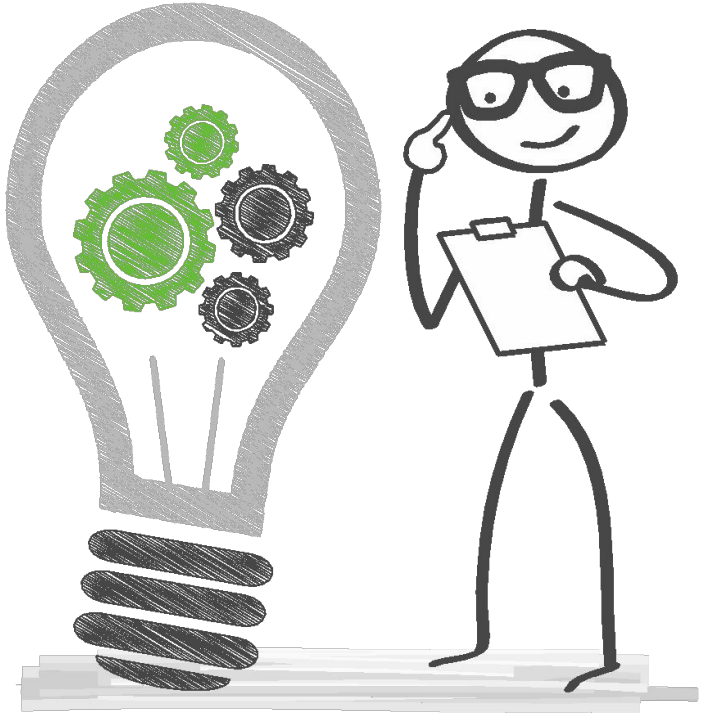


При работе с **at most once** доставкой необходимо убедиться, что мы можем потерять часть сообщений.

Что в итоге?

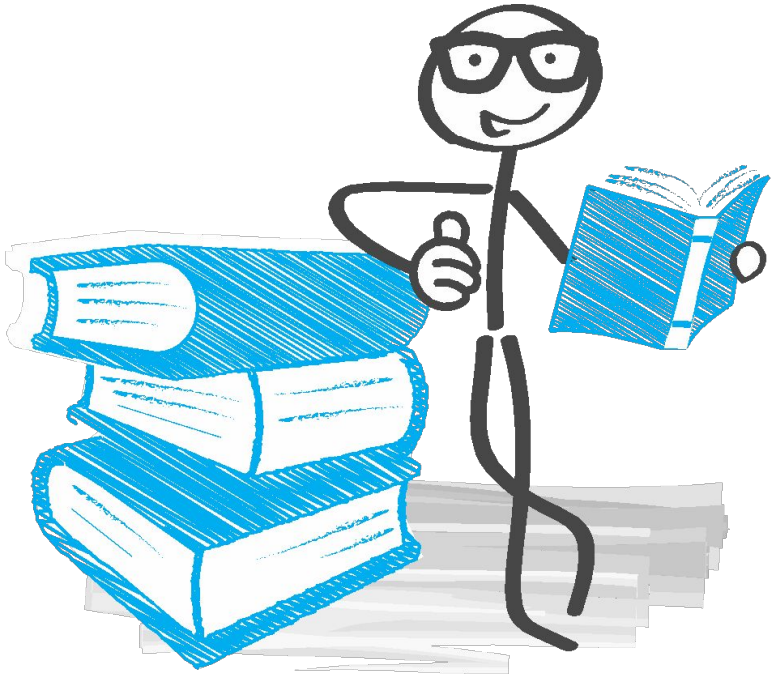


О чем думать



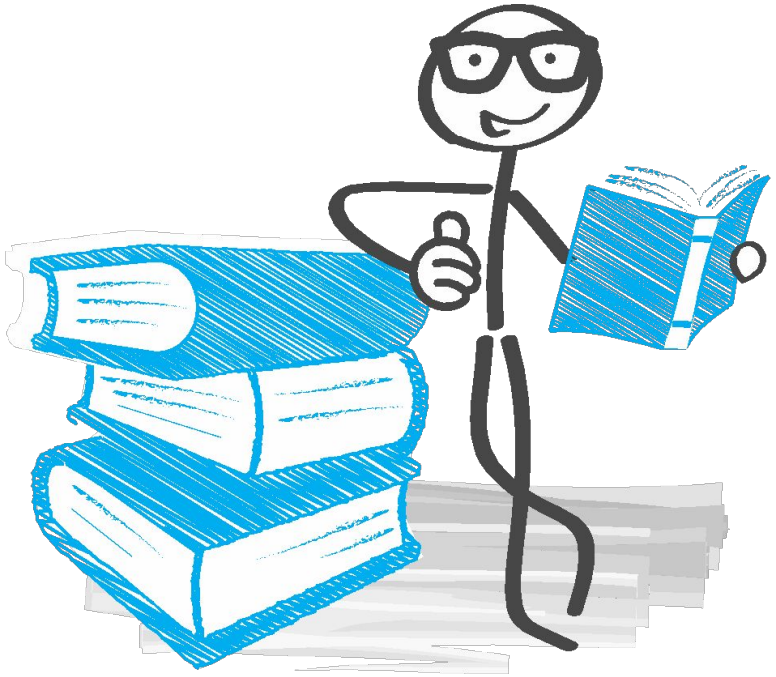
- Выбор модели потребления
- Какие гарантии доставки нужны
- Обработка ошибок и невалидных сообщений
- Идемпотентность сервисов

Что дальше?



- Порядок отправки и обработки сообщений
- Персистентность сообщений
- Брокер как распределенная система

Дополнительные материалы



https://t.me/another_api

Мои контакты

@and_burakov

@another_sa - мой канал об анализе и архитектуре