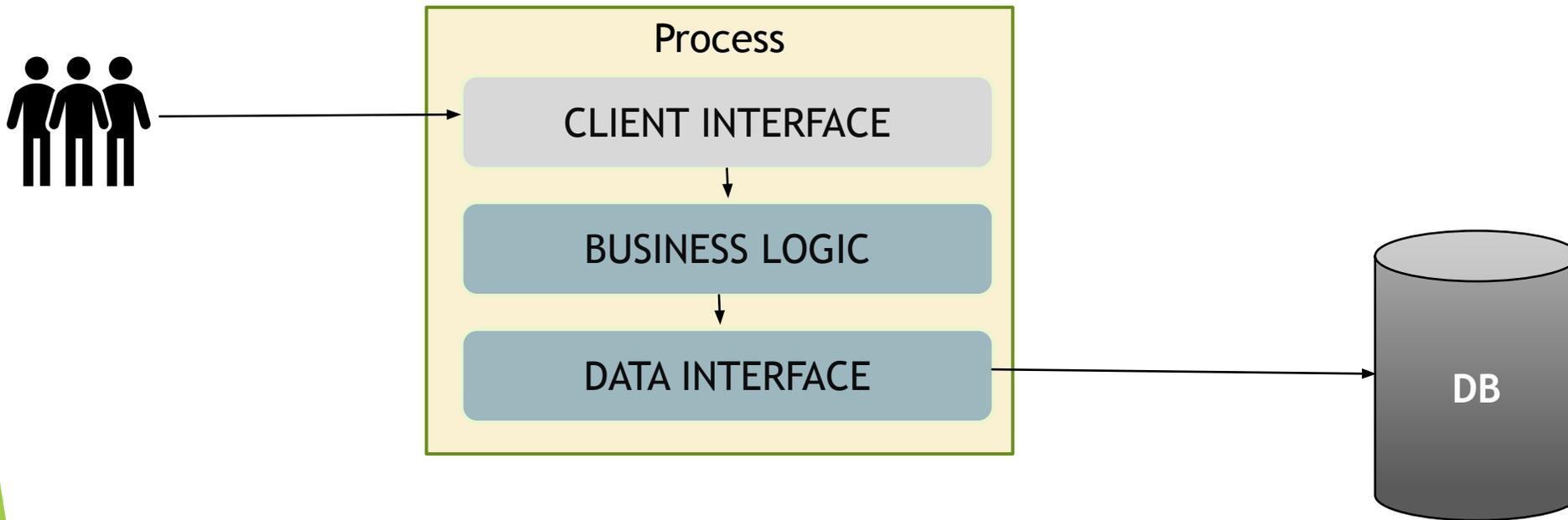


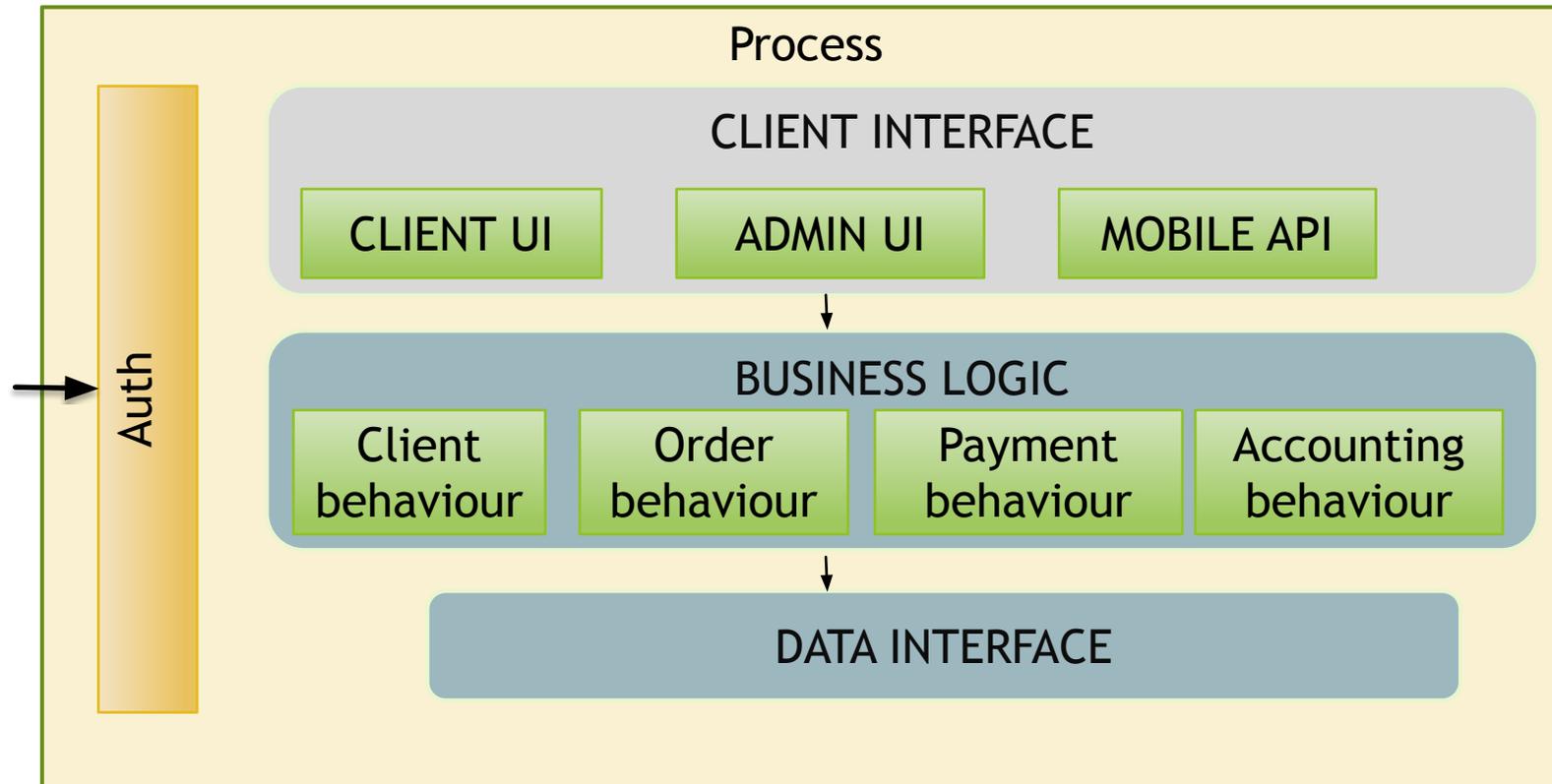
# Микросервисная архитектура

# Монолитное решение



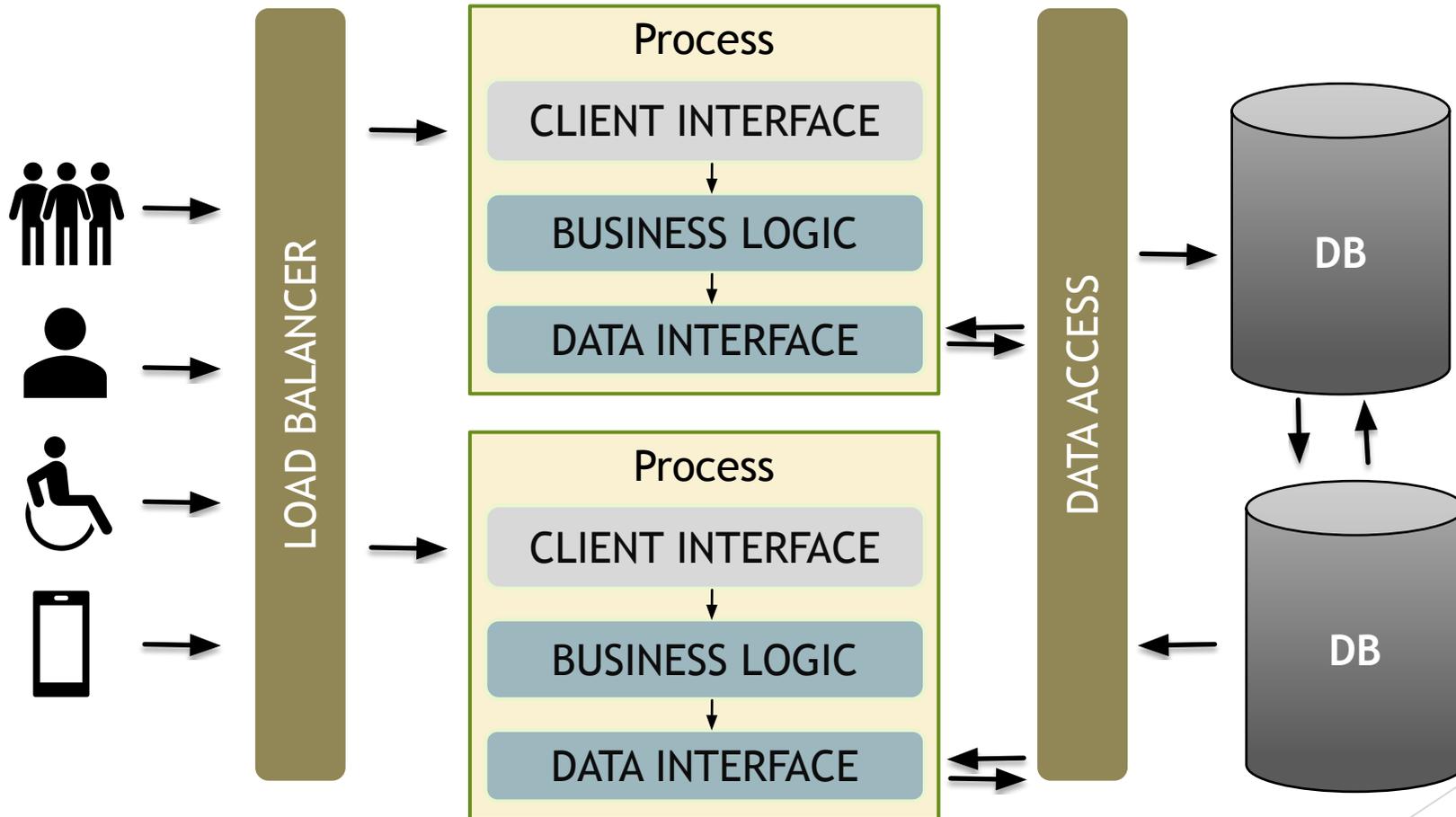
# Монолитное приложение

На пути к успеху



# Монолитное приложение

Успех неминуем



# Микросервисное решение

## □ Инфраструктура приложения

- Разработка
- Развертывание и выполнение
- CI/CD (непрерывная интеграция и доставка)

## □ Архитектура приложения

- Определение сервиса
- Подходы к конфигурированию
- Коммуникации между сервисами
- Данные
- Внешние API приложения
- Логирование и мониторинг
- Тестирование

# Микросервисное решение

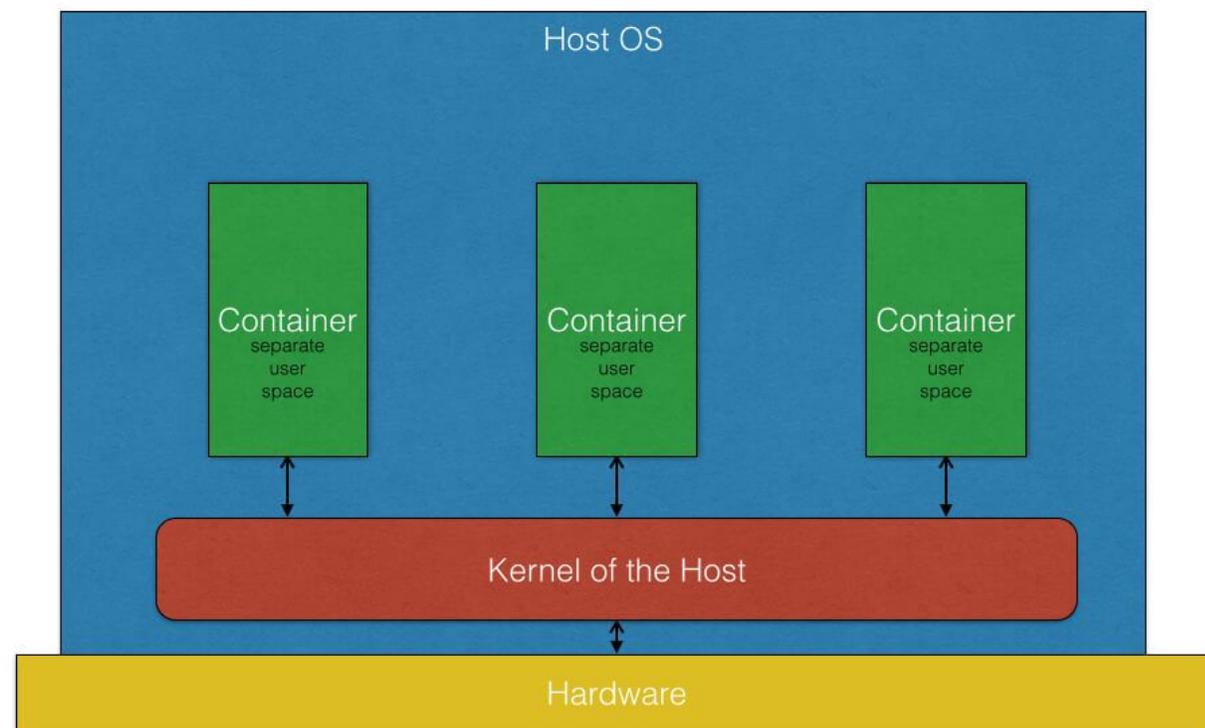
задачи уровня инфраструктуры

- Надежность и отказоустойчивость
- Гибкость и масштабируемость
- Безопасность
- Скорость изменений и CI/CD
- Адекватная стоимость владения

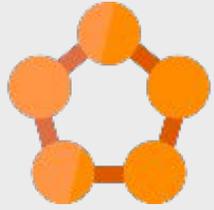
# Микросервисное решение

## Контейнеризация сервисов

- ▶ Гибкая среда
- ▶ Повышенная производительность
- ▶ Управление версиями
- ▶ Переносимость среды вычислений
- ▶ Стандартизация
- ▶ Безопасность

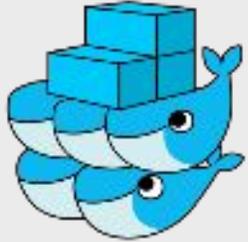


Operating System/Container Virtualization



# Микросервисное решение

Оркестрация приложения

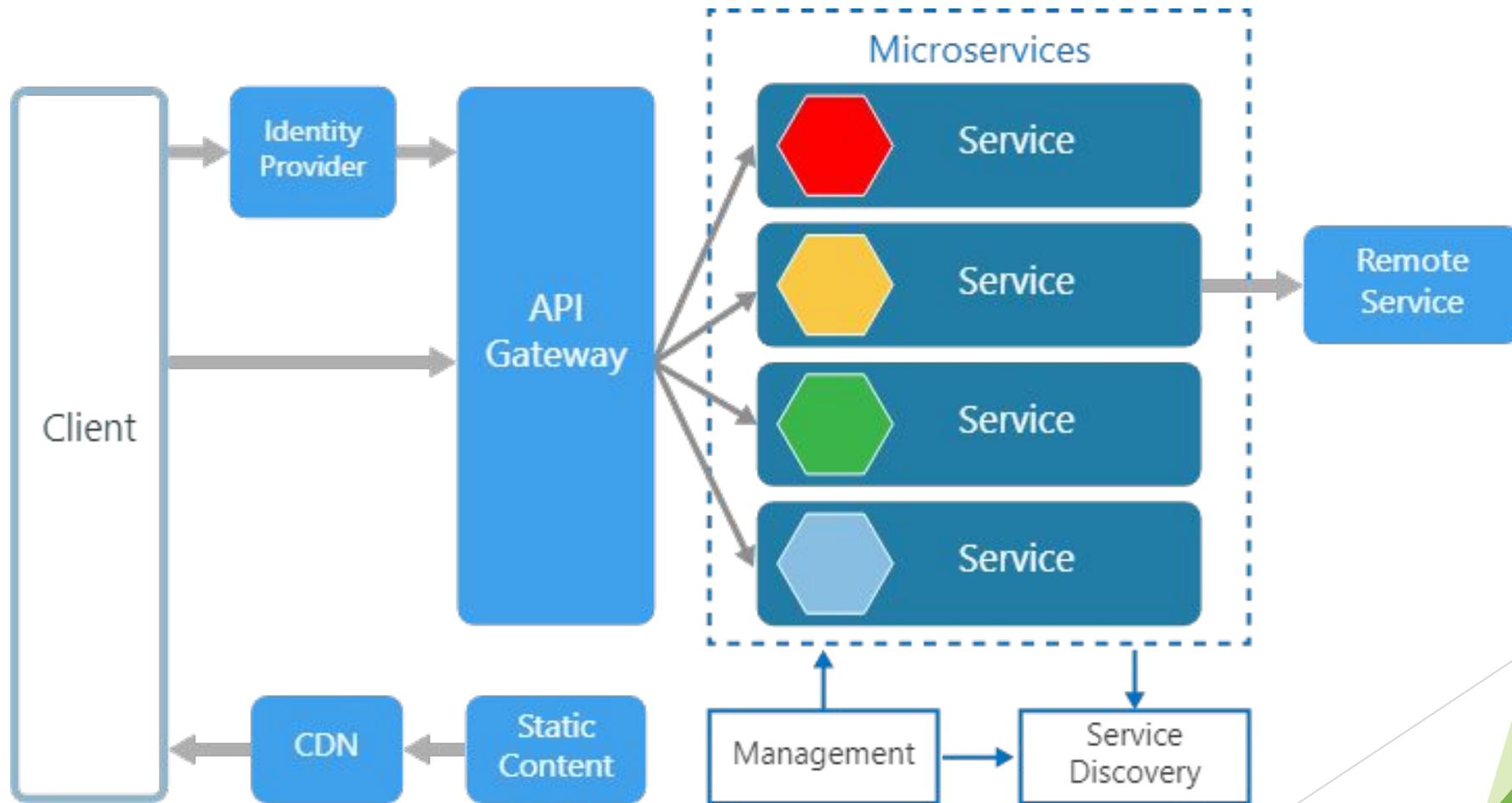


- ▶ Автоматизация развертывания
- ▶ Эффективное использование ресурсов кластера
- ▶ Автоматизация доставки новых версий
- ▶ Обработка отказов



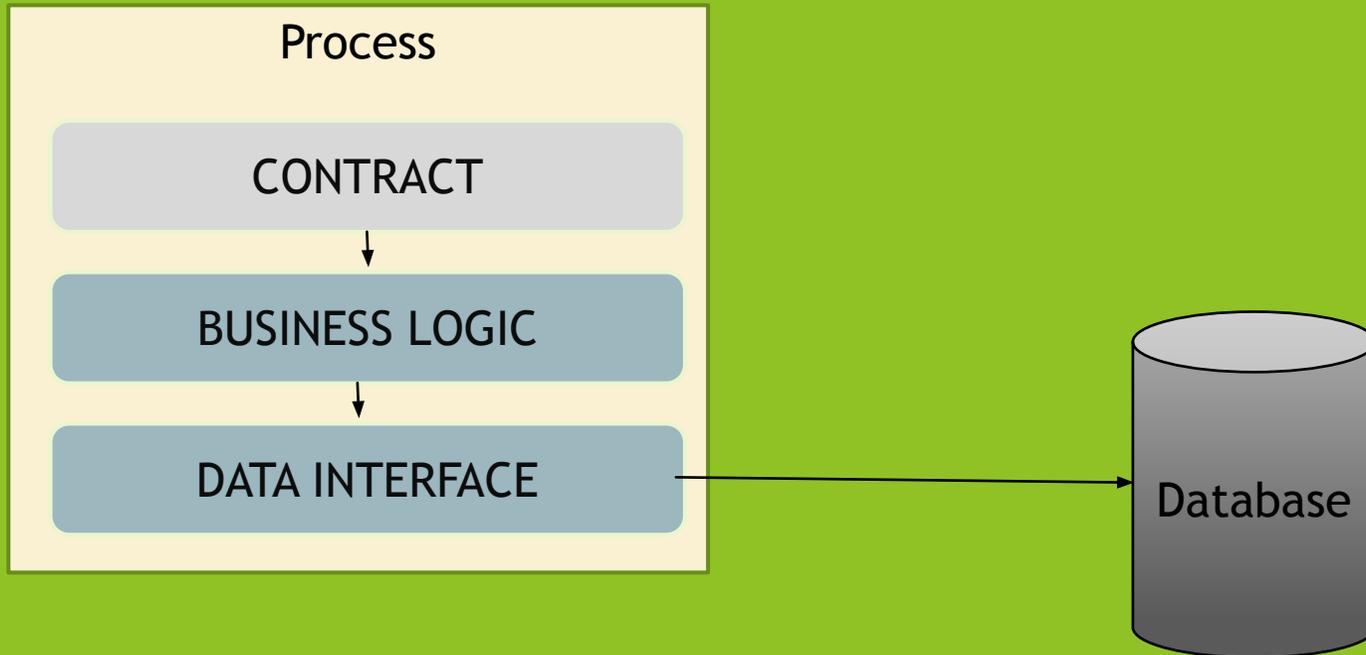
# Микросервисное решение

Архитектура приложения



# Микросервисное решение

Архитектура сервиса приложения



# Микросервисное решение

Определение сервиса

## 1. Определение границ сервиса

Сервис должен:

- оставаться достаточно маленьким
- быть слабо связан с другими сервисами
- отвечать принципу SRP

## □ Размещение конфигурации

- Конфигурация на сервис
- Конфигурация на приложение

## □ Управление конфигурационными настройками

- Обновление настроек без простоя сервиса

# Микросервисное решение

Конфигурирование приложения

## Способы коммуникации

### ▶ Синхронные вызовы

- Вызов удаленных процедур (RPC)
- REST

### ▶ Асинхронные сообщения

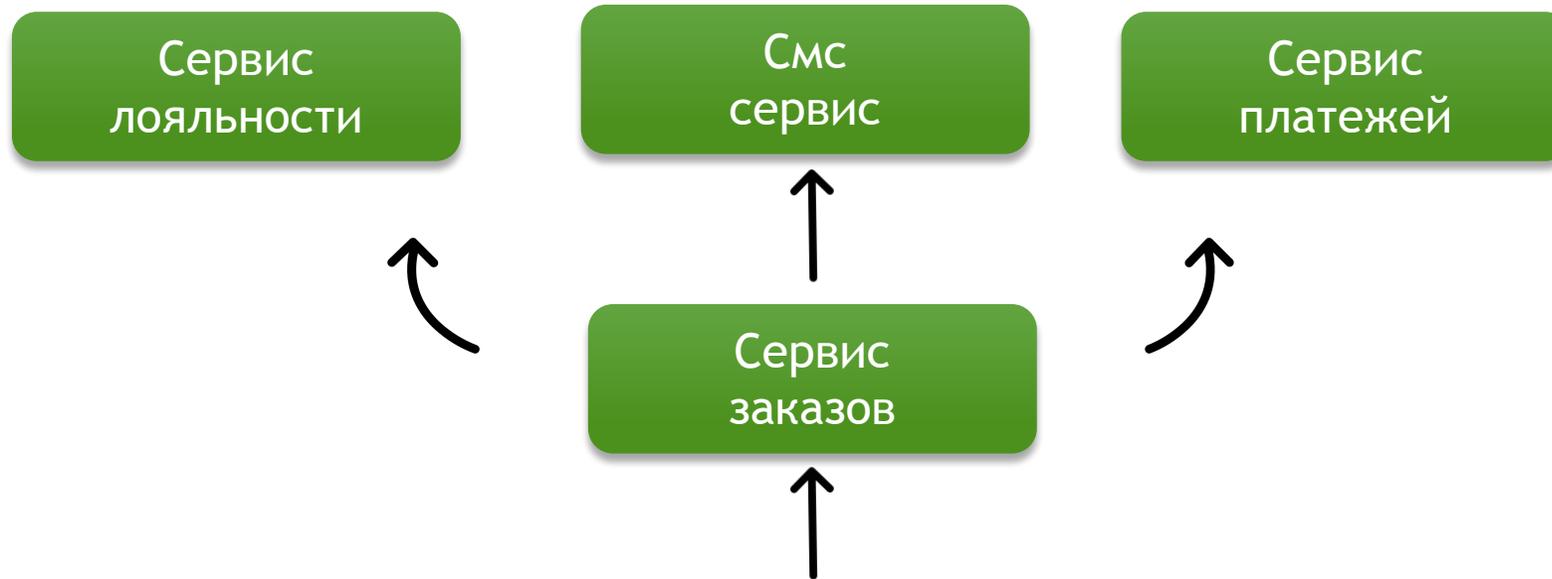
- Брокер сообщений
- Веб хуки

# Микросервисное решение

Коммуникации между сервисами

# Микросервисное решение

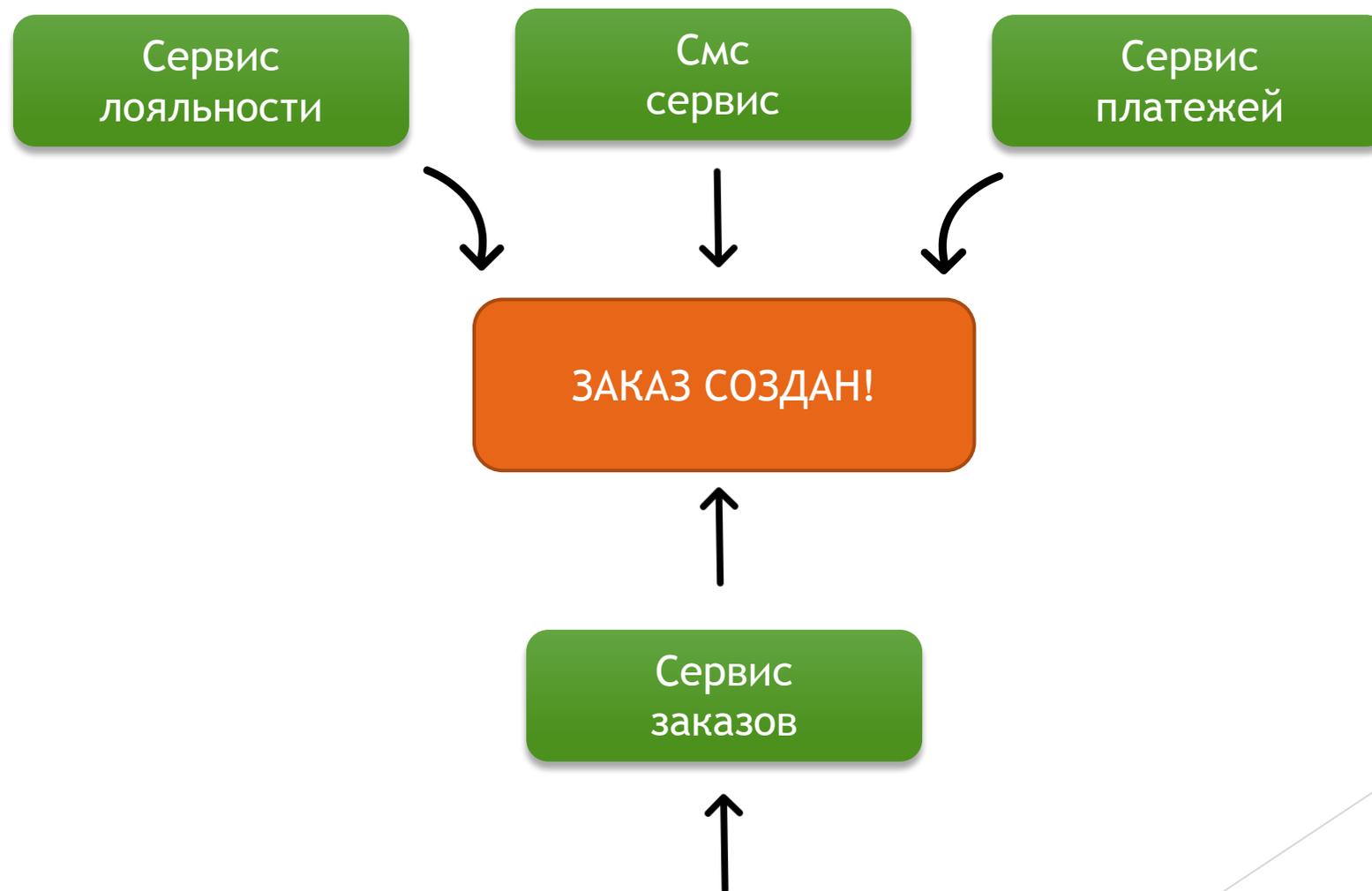
*Оркестрируемые коммуникации*



1. Создать заказ на приобретение товара
2. Снять деньги с карты клиента
3. Отправить клиенту смс
4. Учесть покупку в программе лояльности клиента

# Микросервисное решение

*Хореография коммуникаций*



## Организация данных

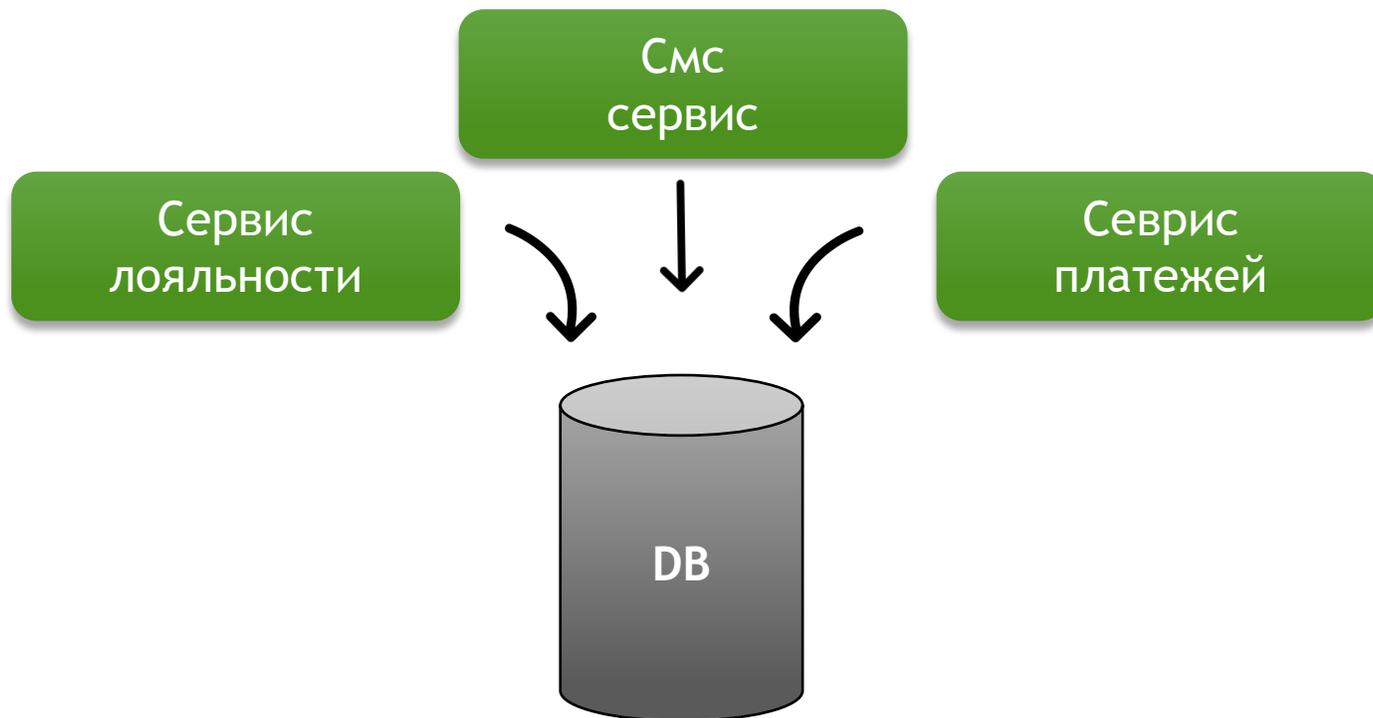
- ❖ База данных на сервис
- ❖ Общая база данных

## Микросервисное решение

Архитектура данных

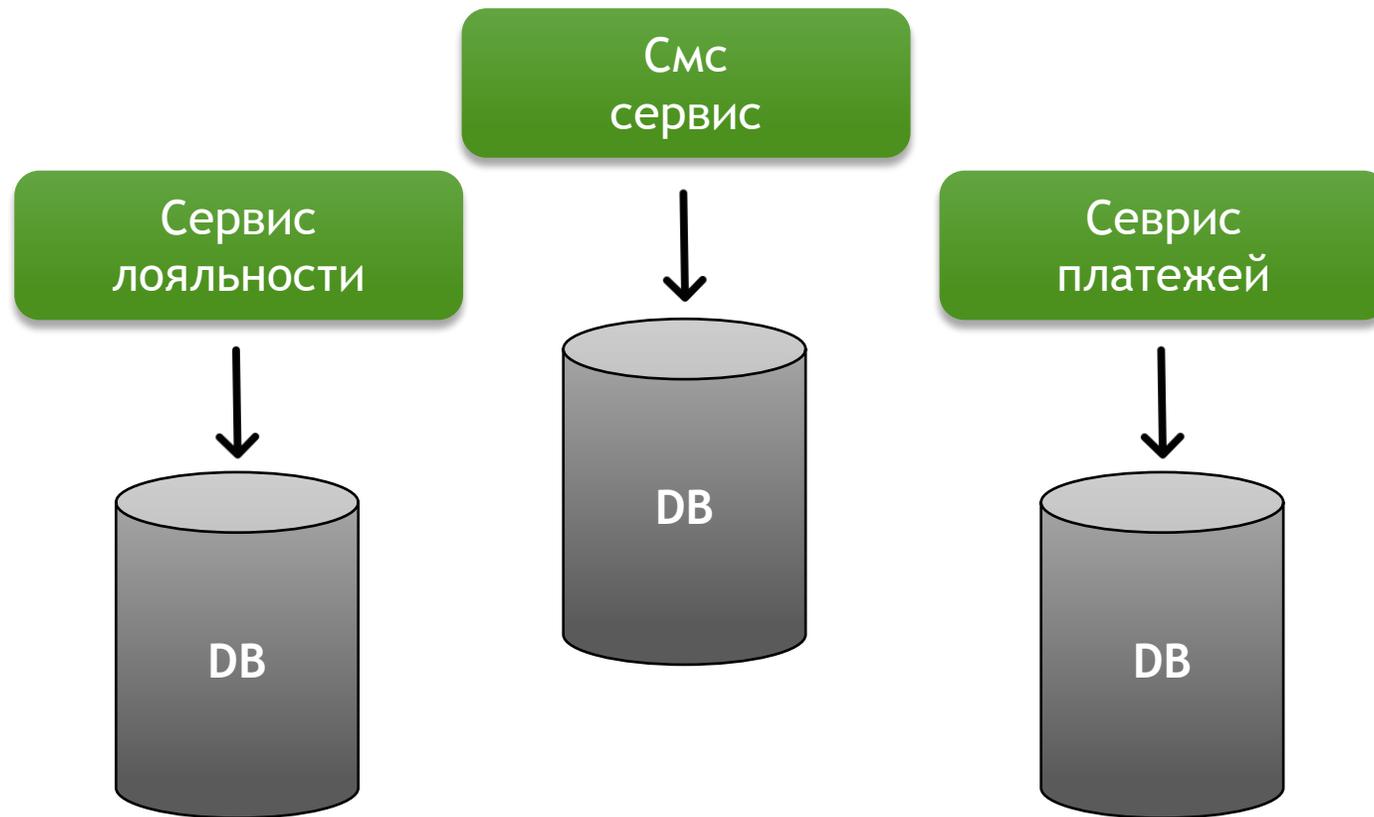
# Микросервисное решение

*Общая база данных*



# Микросервисное решение

*База данных на сервис*



01

API сервисов

02

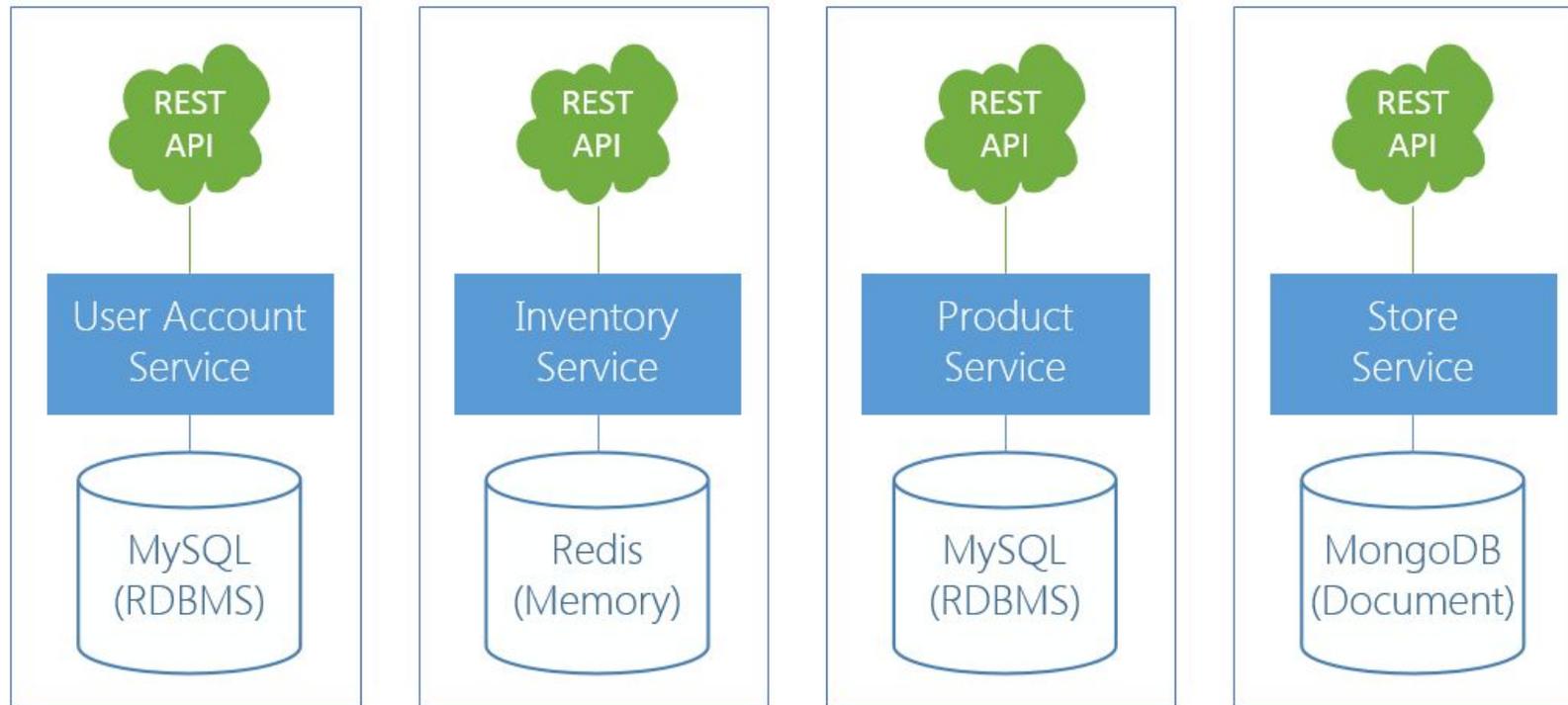
API сервис для клиента

# Микросервисное решение

Публичные интерфейсы приложения

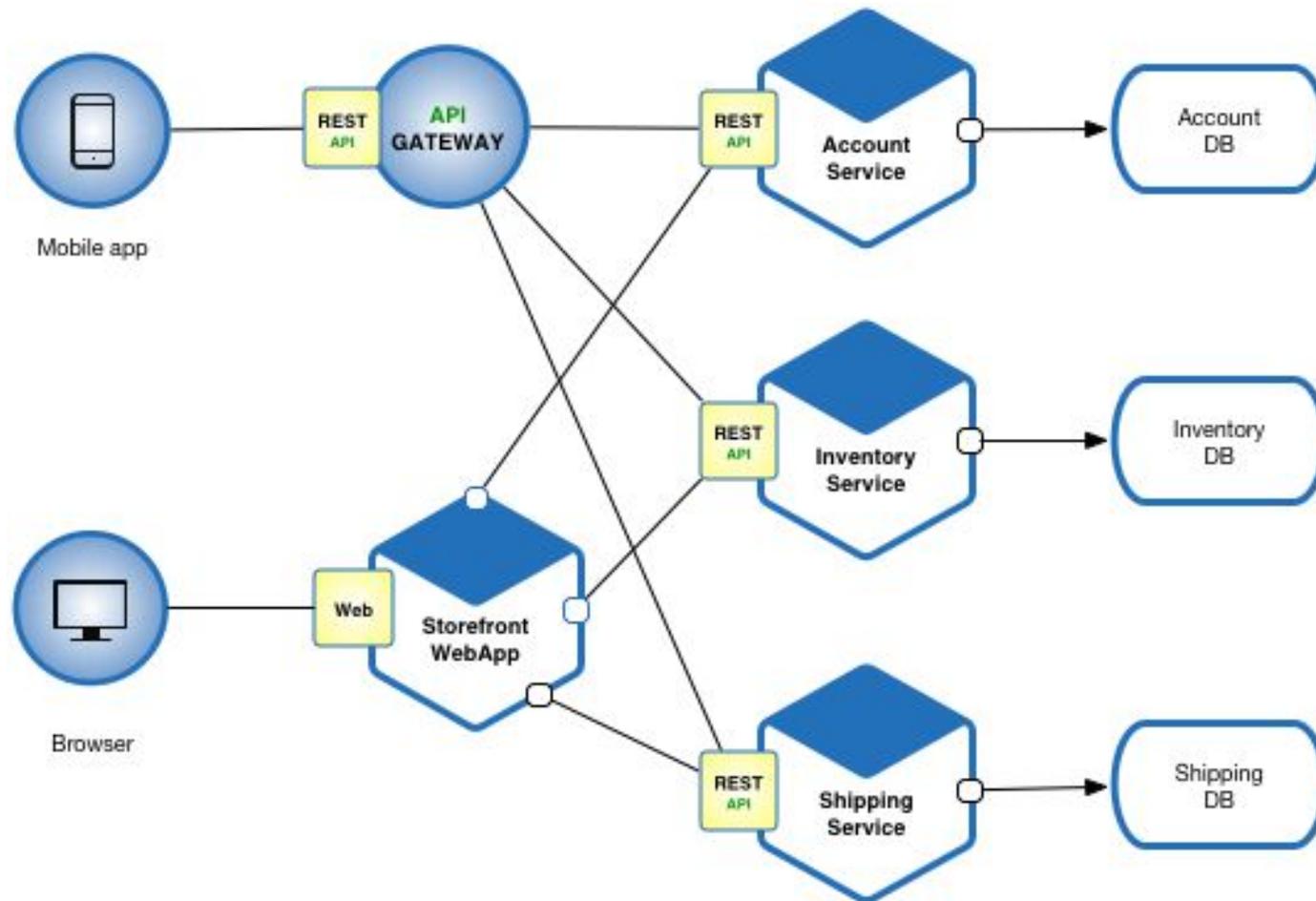
# Микросервисное решение

*API сервисов*



# Микросервисное решение

*API для клиента*



- ❖ Метрики приложения
- ❖ Обнаружение проблем
  - Превентивное
  - Фактическое
- ❖ Аудит
  - Выполнение
  - Трассировка

# Микросервисное решение

Логирование и мониторинг

- ❖ Unit-тестирование
- ❖ Тестирование сервисов
  - На основе ожиданий потребителей
  - Со стороны потребителей
- ❖ Интеграционное тестирование

Микросервисное  
решение  
Тестирование

**ИТОГ**

**КОНЕЦ**