

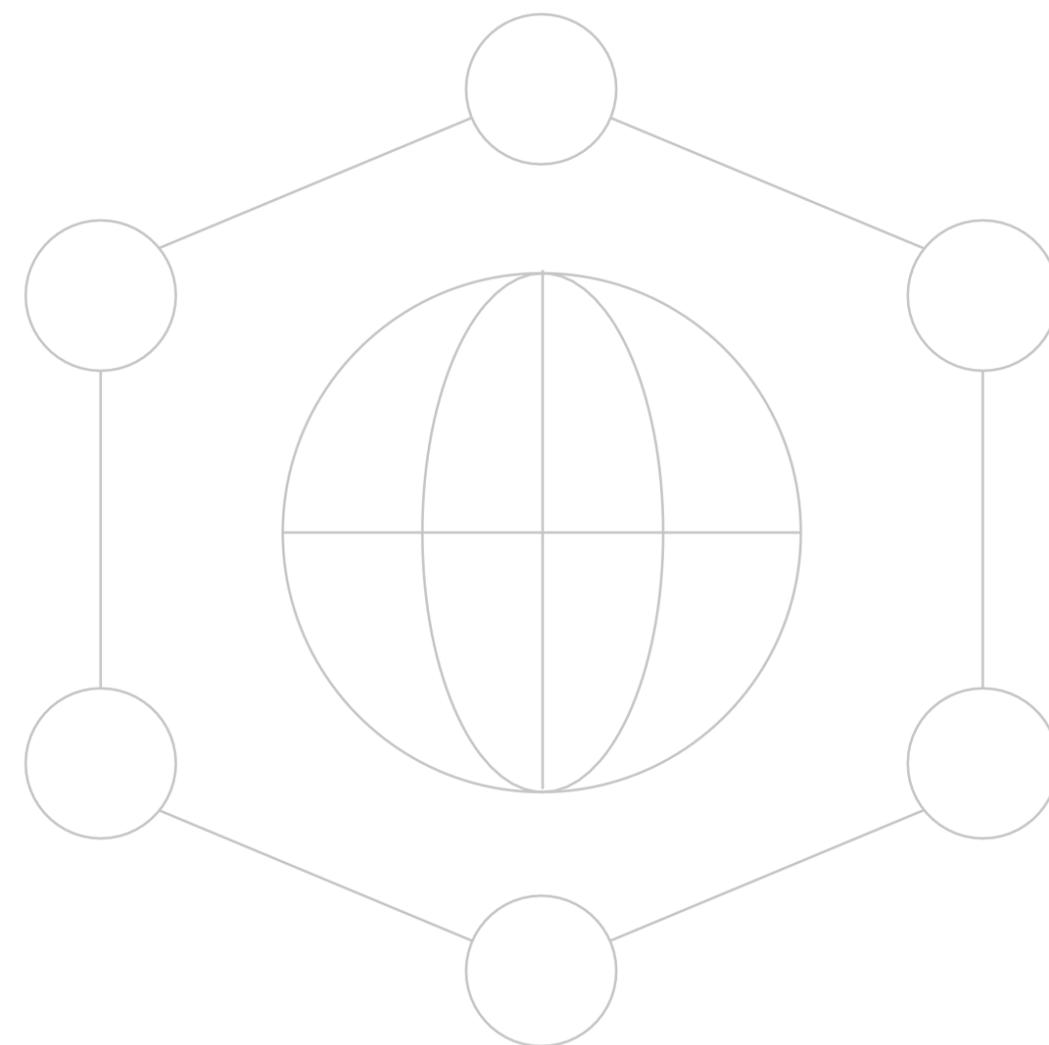
Основные способы построения программных архитектур

Семен Киреков
Java Team Lead
MTS Big Data

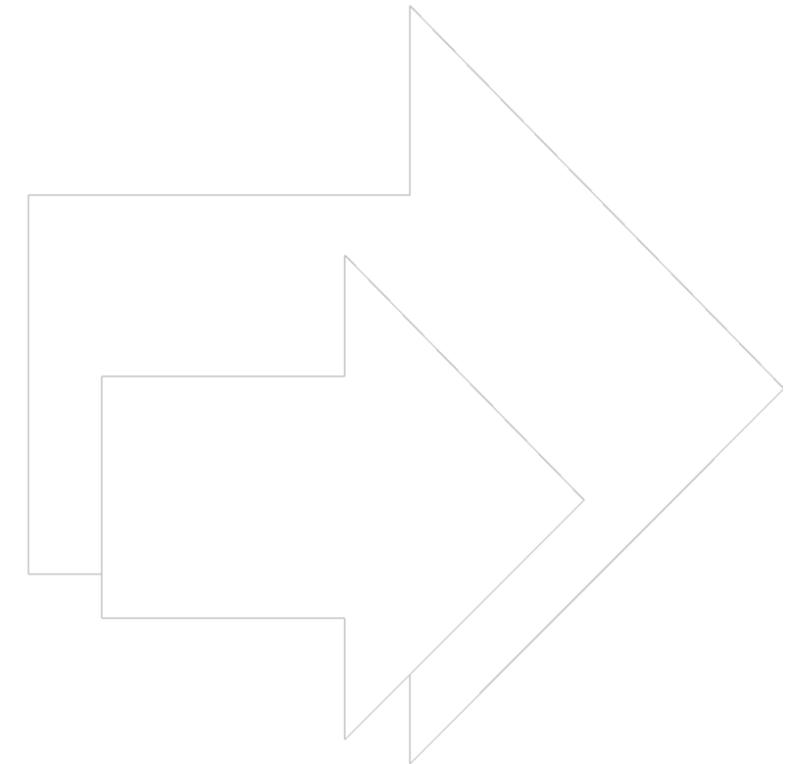


О чем поговорим

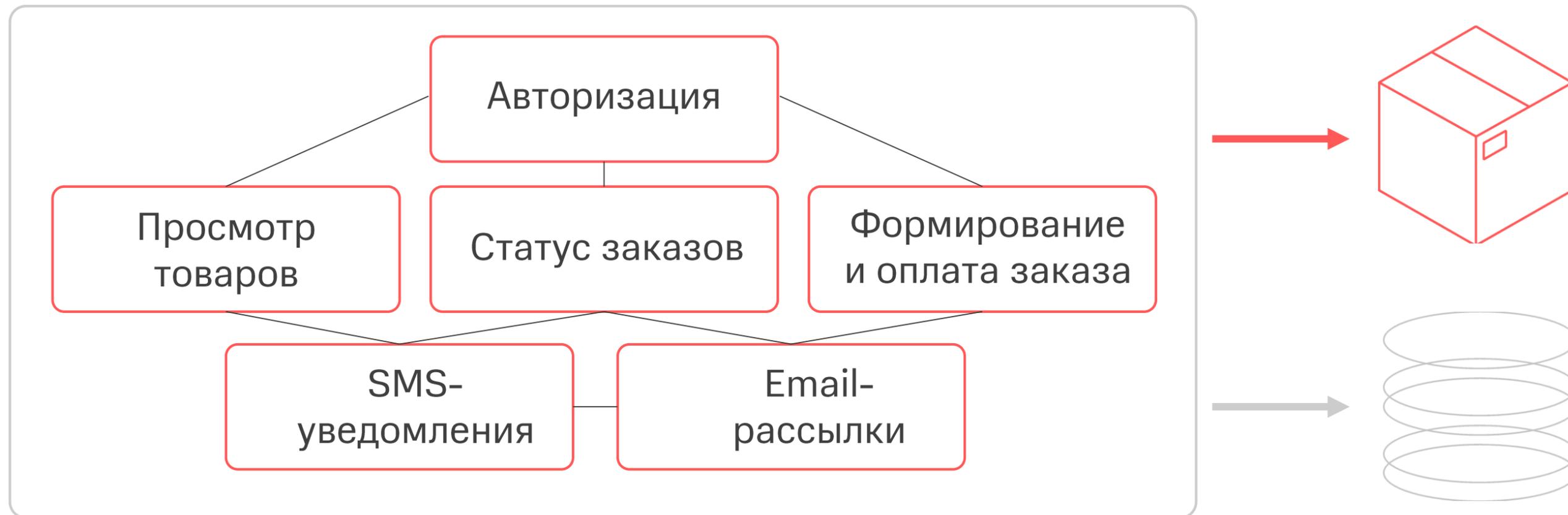
- Монолитная архитектура
- Микросервисная архитектура
- Плюсы и минусы



Монолитное приложение (монолит) —
это приложение, которое доставляется
конечным клиентам в качестве единого
неделимого артефакта.



Интернет-магазин «Моя Книга»



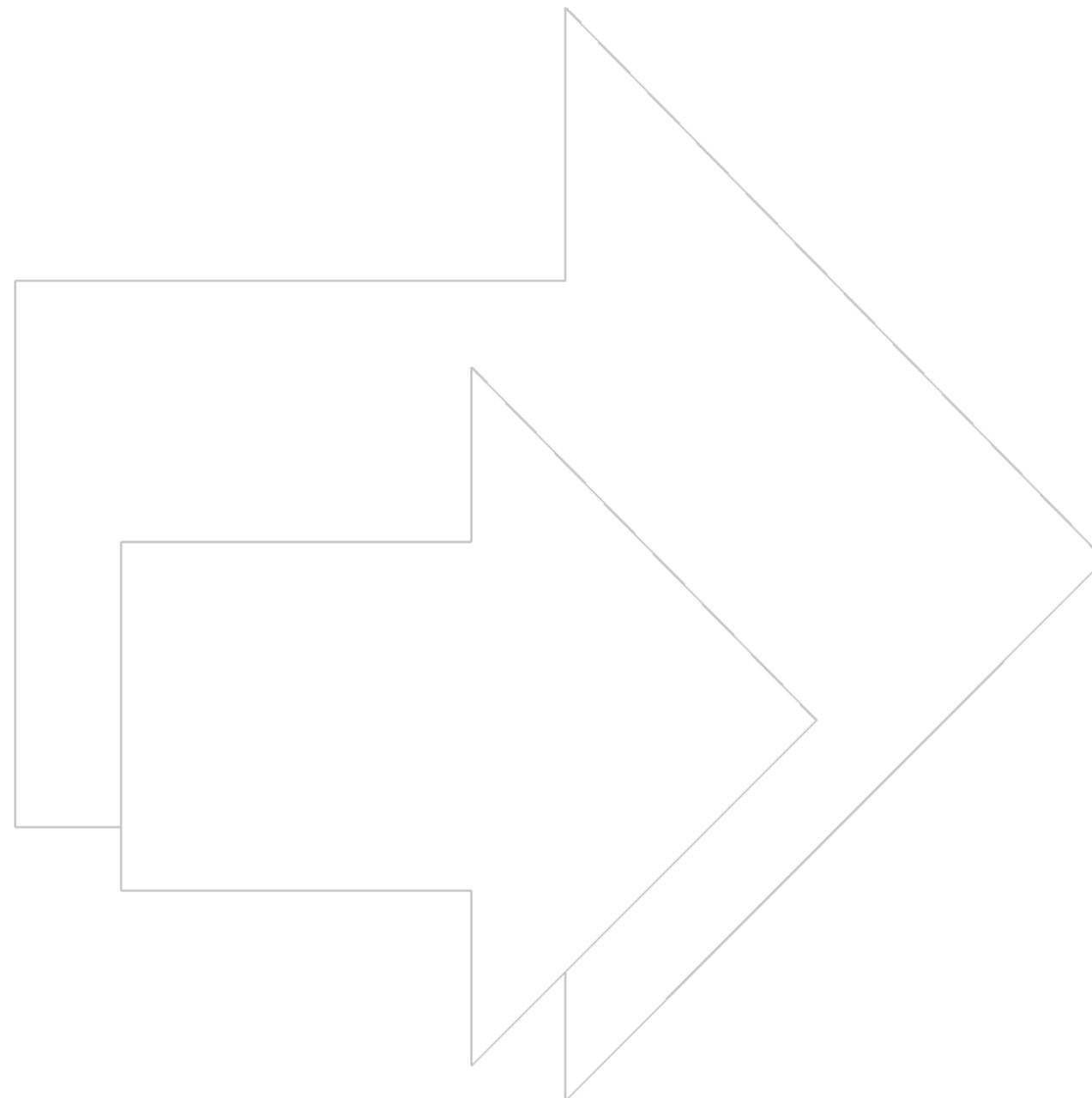
Плюсы монолита

- Простой деплой
- Низкий time-to-market
- Согласованность данных
- Инкапсуляция зависимостей
- Мониторинг
- Единые правила контрибьютинга

Минусы монолита



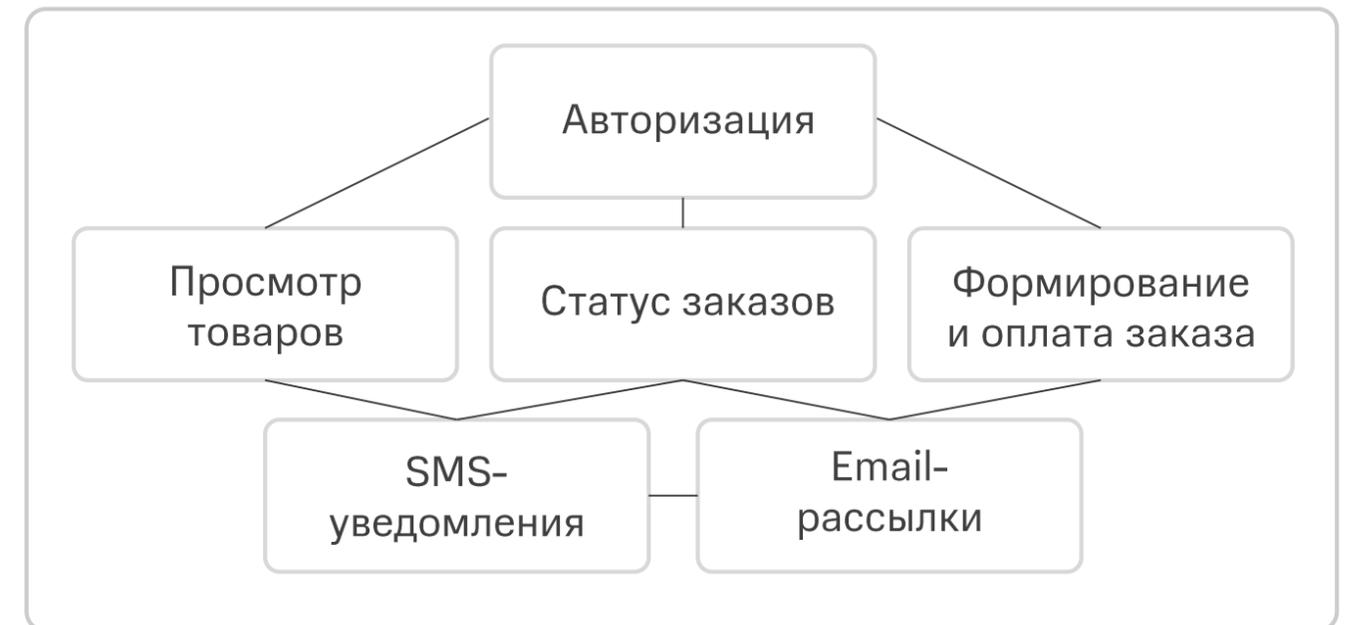
Горизонтальная
масштабируемость



**32 Гб RAM,
4 Cores**



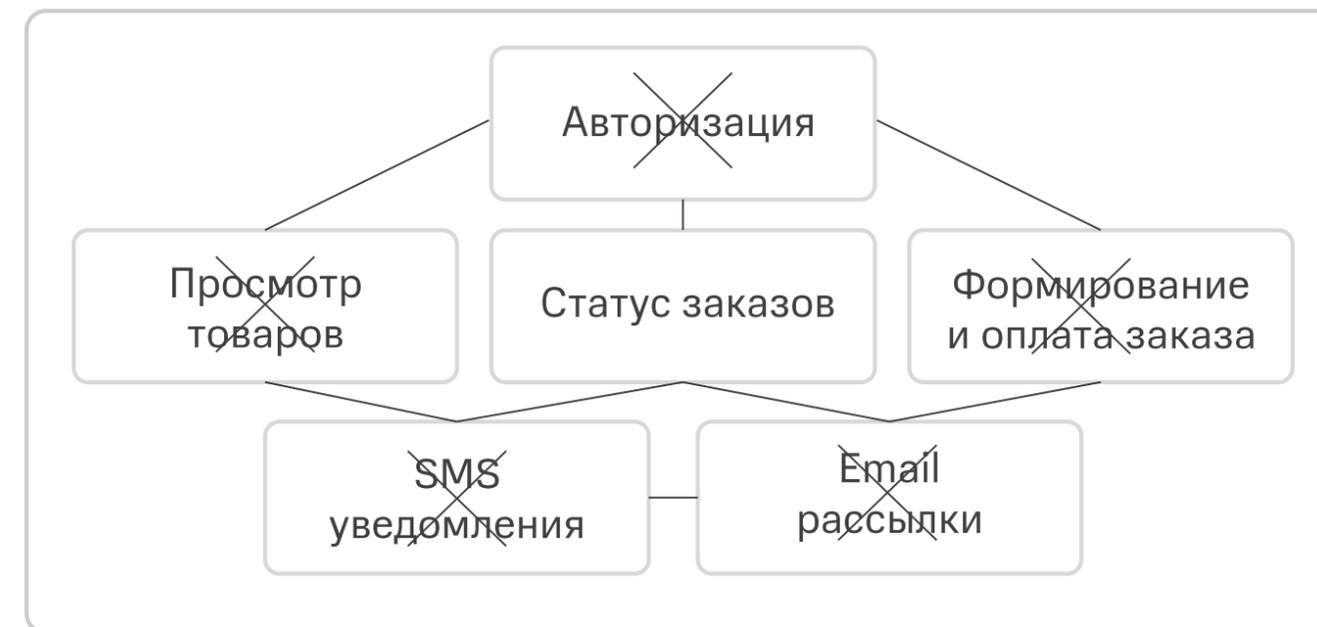
**32 Гб RAM,
4 Cores**



**32 Гб RAM,
4 Cores**

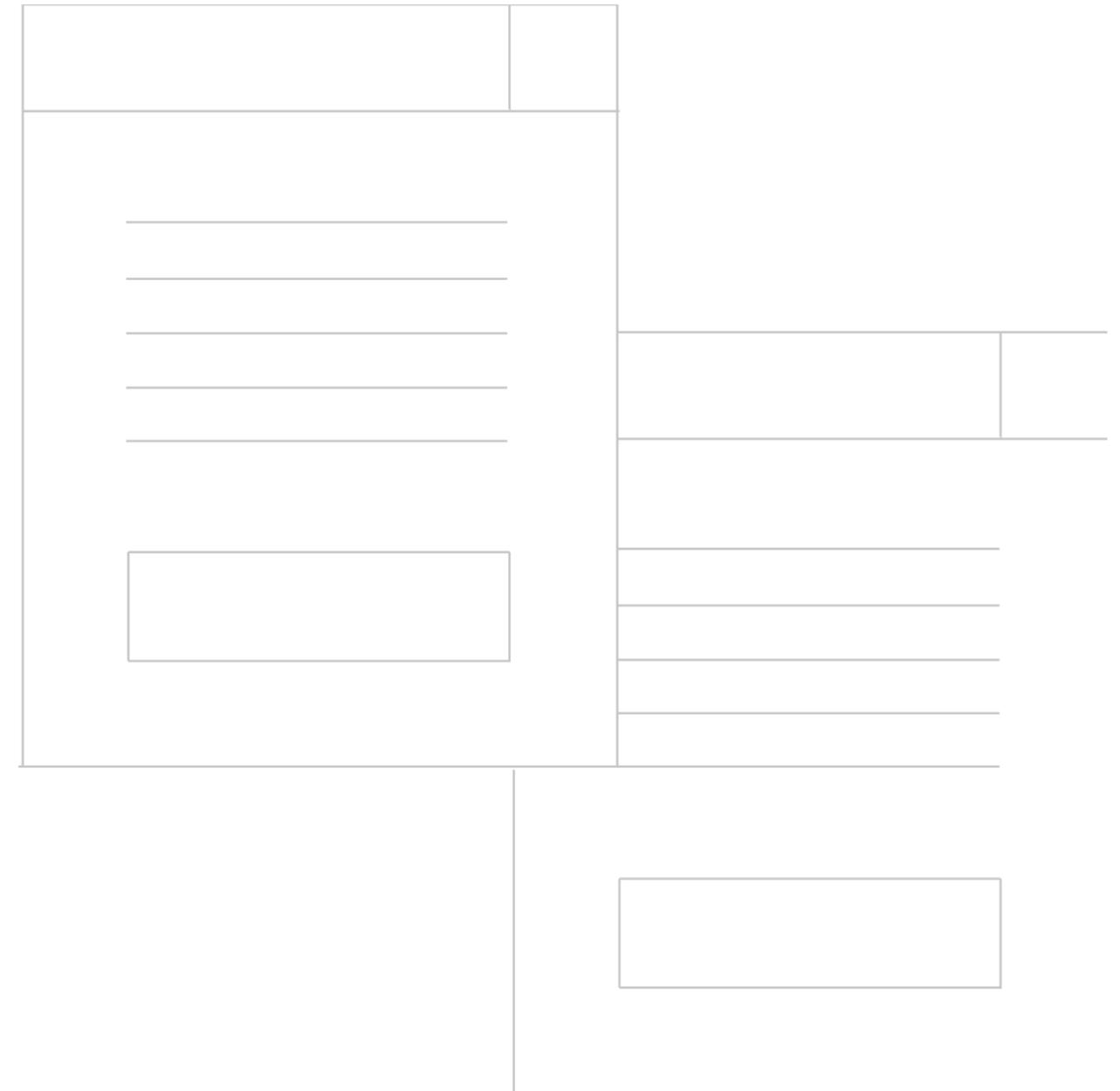


**32 Гб RAM,
4 Cores**



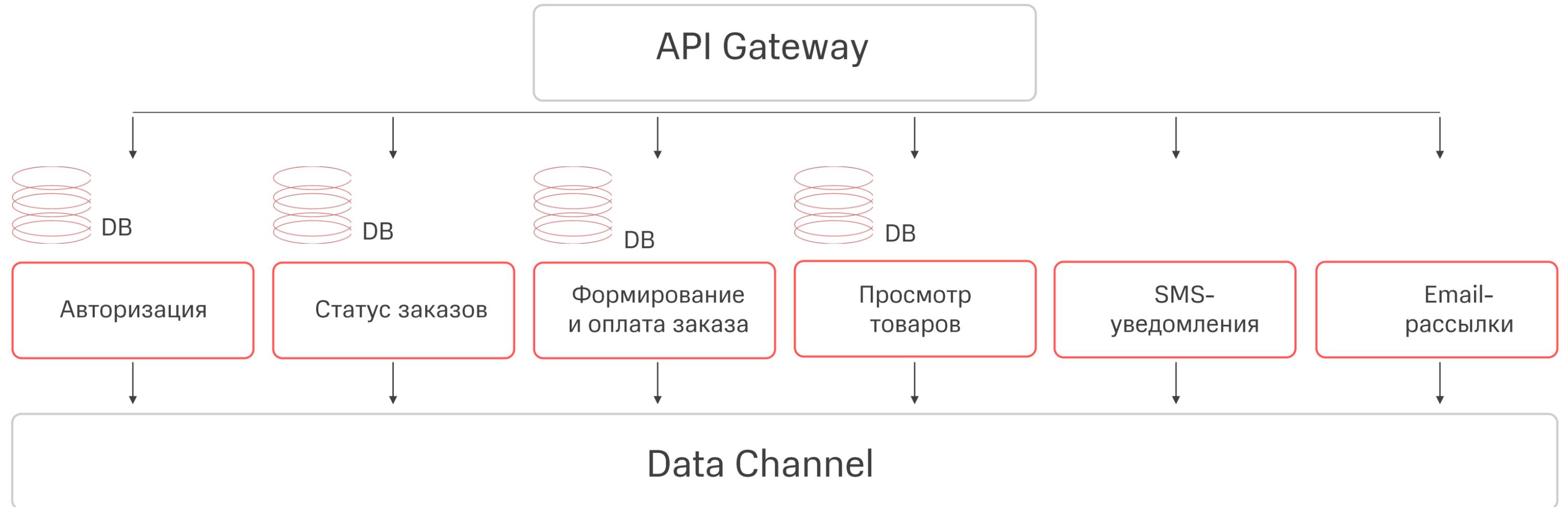
Минусы монолита

- Горизонтальная масштабируемость
- Низкая отказоустойчивость
- Спагетти-код
- Сложность внедрения новых технологий
(много legacy-кода)



Микросервисная архитектура — вариант сервис-ориентированной архитектуры ПО, направленный на взаимодействие на столько это возможно небольших, слабо связанных и легко изменяемых модулей — микросервисов.

Интернет-магазин «Моя Книга»

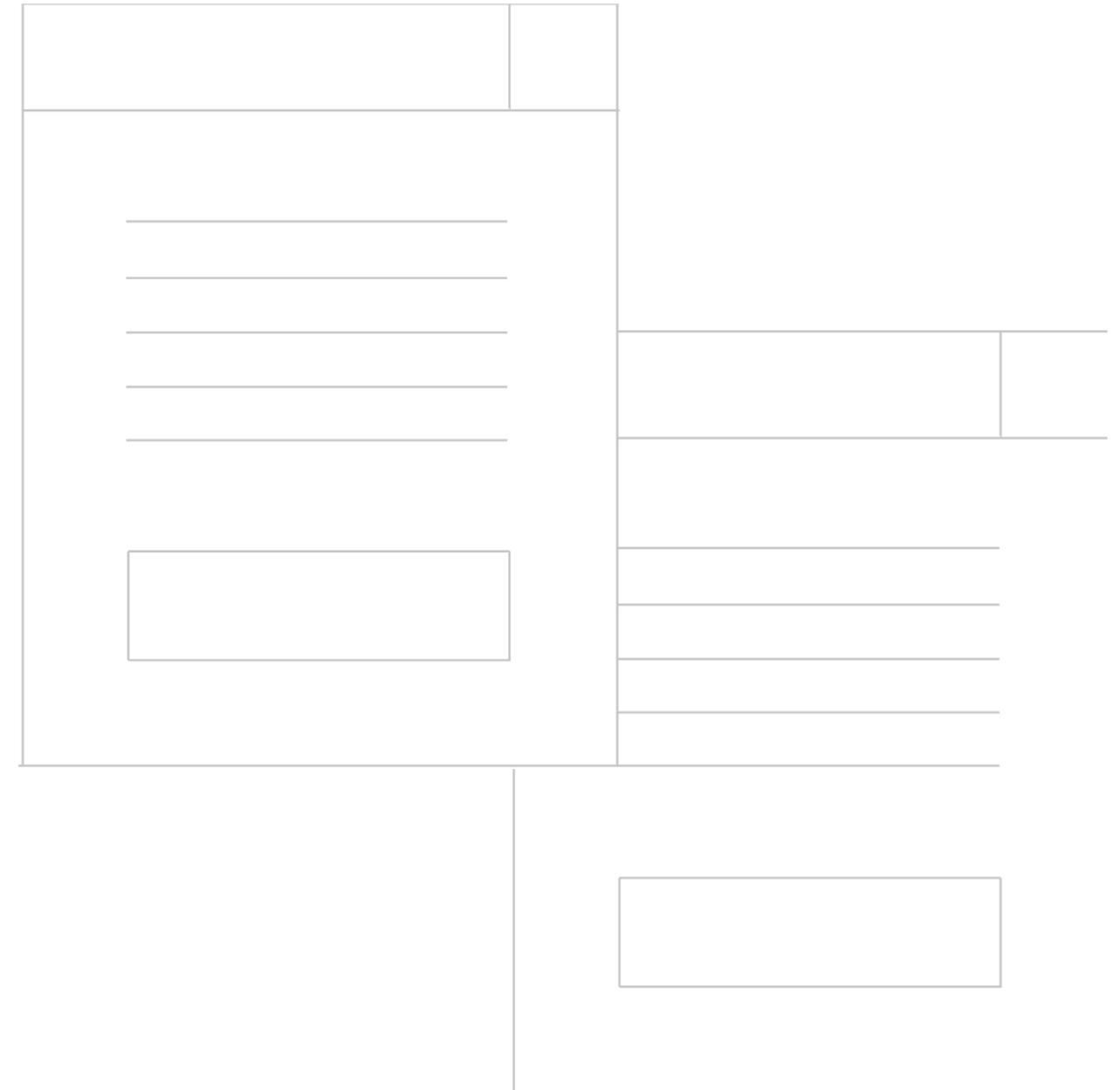


Плюсы микросервисов

- Хорошая горизонтальная масштабируемость
- Обеспечение отказоустойчивости
- Независимая деградация
- Легче внедрять новые технологии
- Большая команда быстрее разрабатывает
- Независимые релизы

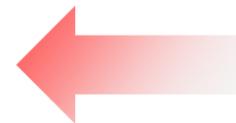
Минусы микросервисов

⚠ Версионирование API



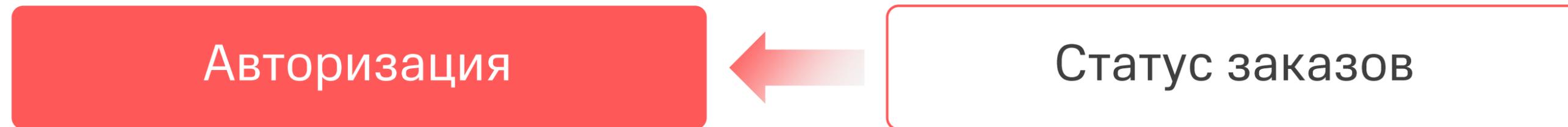
Версия 1

Авторизация



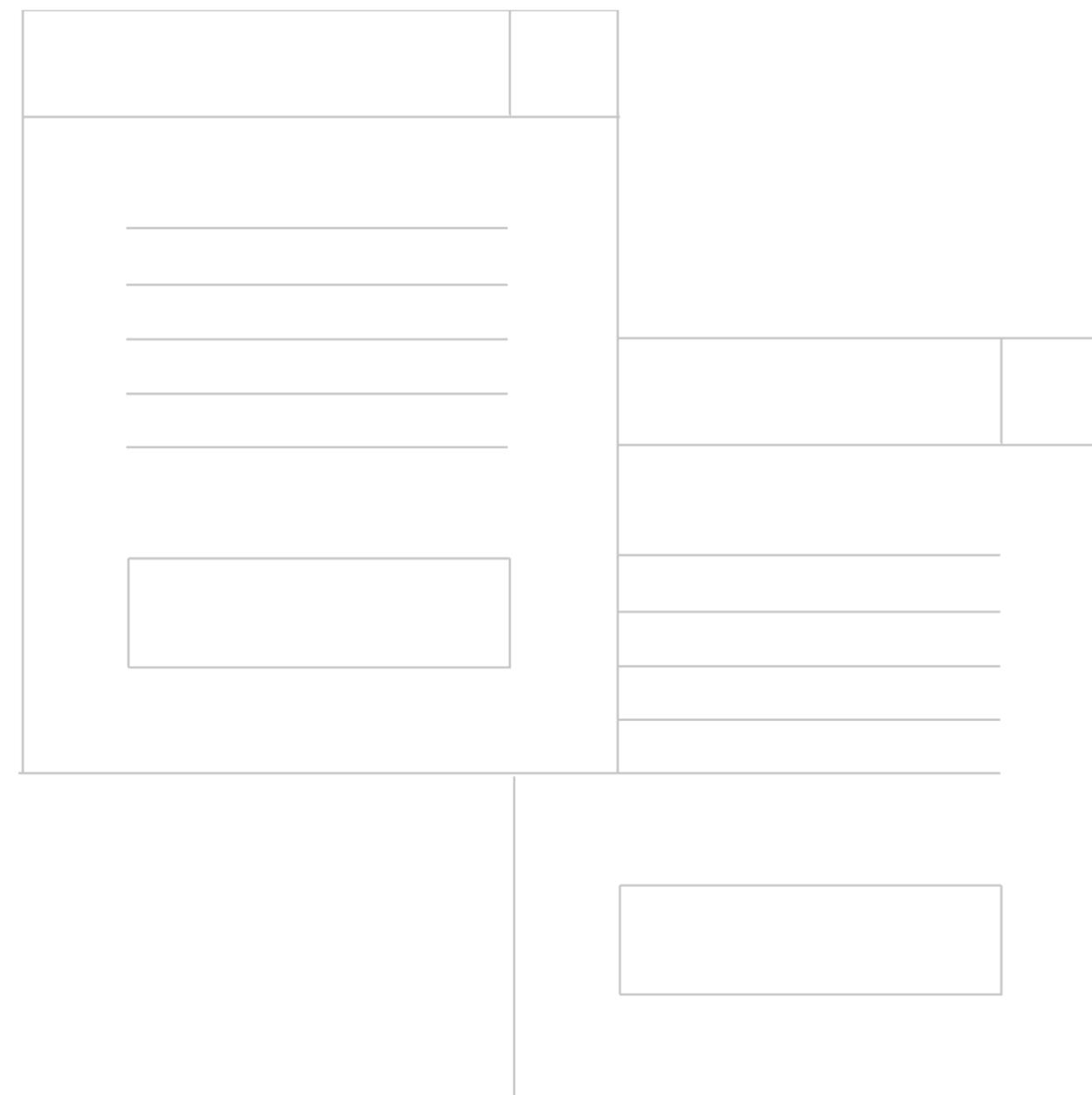
Статус заказов

Версия 2



Минусы микросервисов

- Версионирование API
(поддержка нескольких версий)
- Сложный деплой
- Согласованность в конечном счете
- Обработка ошибок
- Логирование
Мониторинг
- Тестирование



Выводы

Монолит

Быстрый старт, сложности
в перспективе

Микросервисы

Сложности на старте,
плюсы в перспективе

Спасибо!

