

Skillbox

# Распределенная архитектура

**Service-Oriented Architecture Style**

**Андрей Гордиенков**

Solution Architect  
ABAX

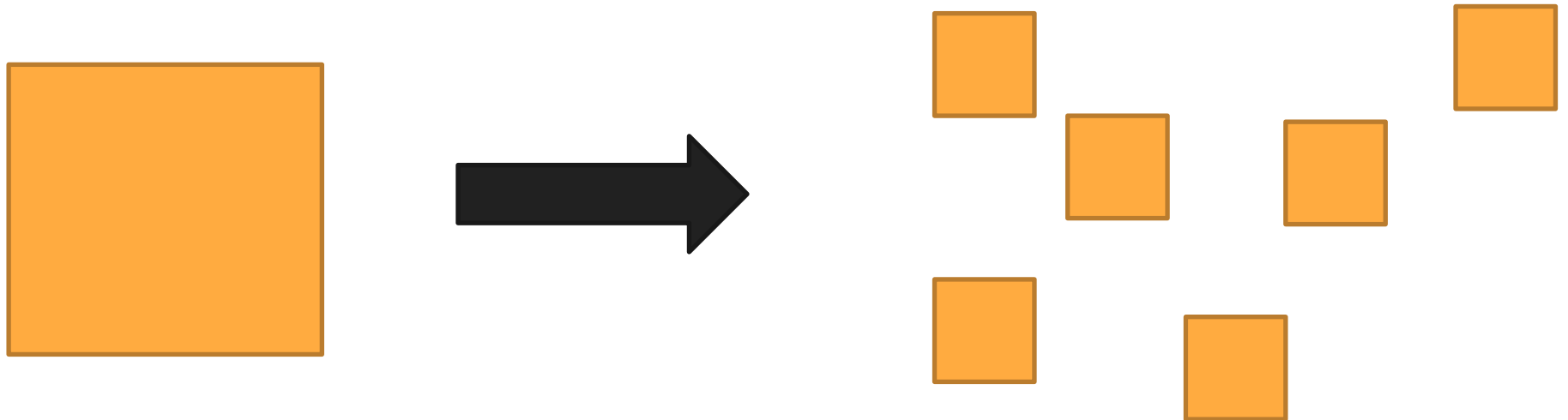
## **В прошлом видео**

- Узнали про «Первое правило» распределенных систем
- Подходы к реализации доменной логики

## **В этом видео**

- Что такое Service-Oriented Architecture
- Понятия и принципы
- Причины сложности

# Важно помнить



- Распределенные системы **всегда** сложнее
- Компоненты **возможно** реализовывать проще

# Что такое сервис-ориентированная архитектура (SOA)?



# Что такое SOA?



# Что такое SOA?

SOA – это модульный подход в разработке приложений, основанный на использовании:

- распределенных,
- слабо связанных,
- заменяемых

компонентов, доступных по сети и предоставляющих четко-определенный формат общения.

# Манифест SOA

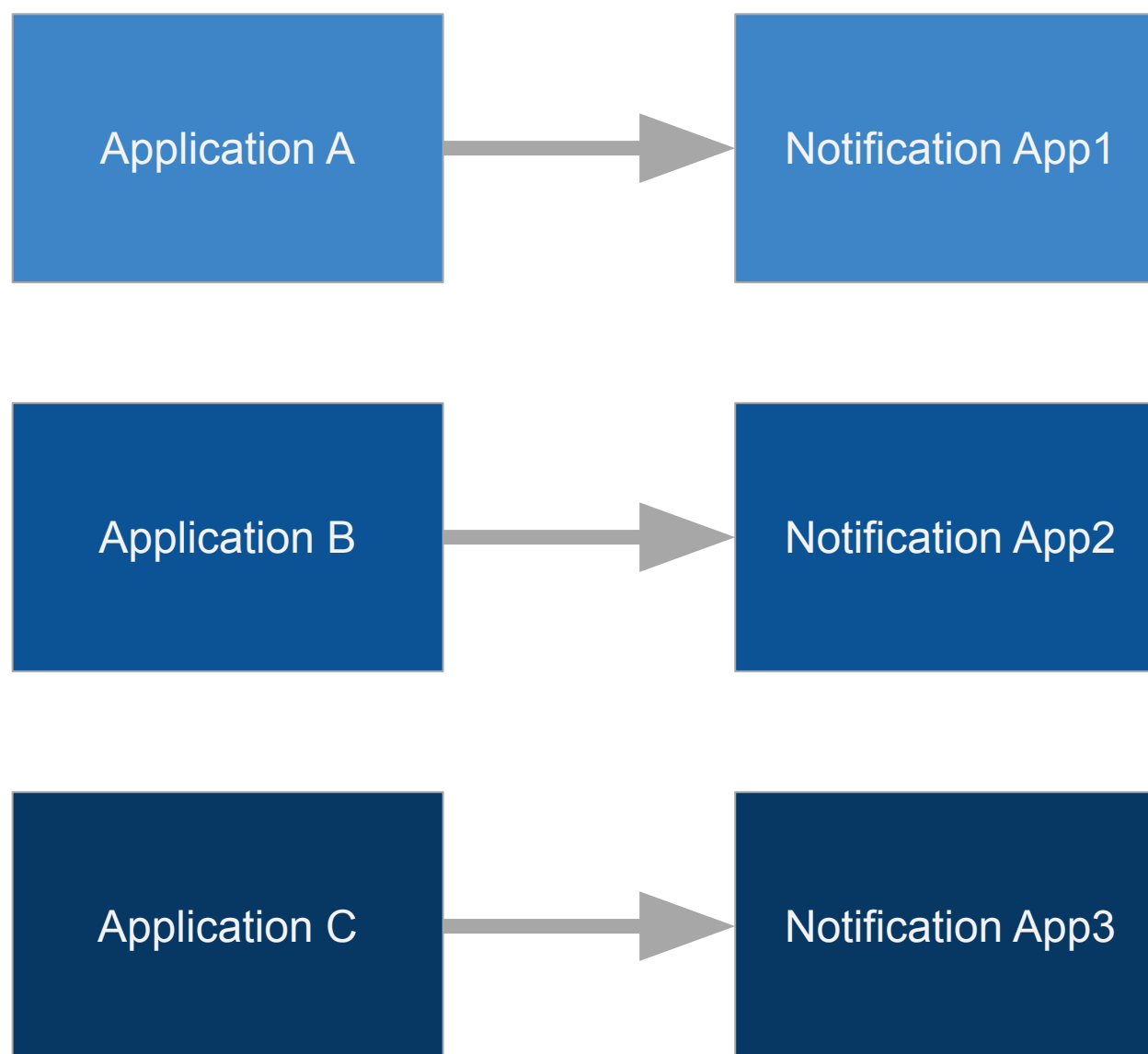
1. Ценность бизнеса важнее технологических средств;
2. Стратегические цели важнее, чем выгода конкретных проектов (иначе говоря, стратегические цели важнее тактических);
3. Внутренняя совместимость важнее, чем точечная интеграция;
4. Общие (распределенные и повторно используемые) сервисы важнее, чем узкоспециализированные;
5. Гибкость важнее, чем оптимизация;
6. Усовершенствование в развитии важнее, чем стремление к изначальному совершенству.

# Манифест SOA

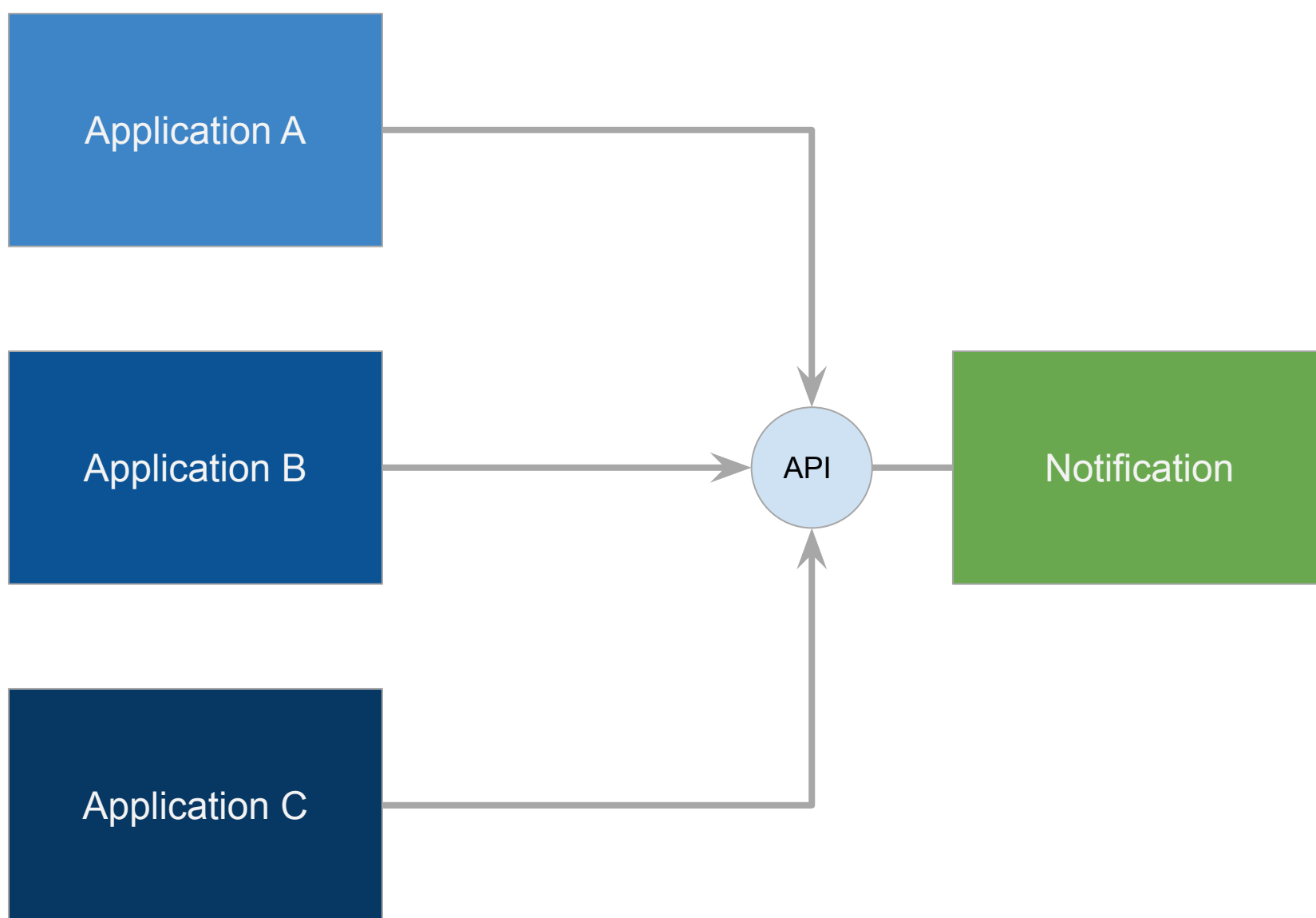
1. Ценность бизнеса важнее технологических средств;
2. Стратегические цели важнее, чем выгода конкретных проектов (иначе говоря, стратегические цели важнее тактических);
3. Внутренняя совместимость важнее, чем точечная интеграция;
4. **Общие (распределенные и повторно используемые) сервисы важнее, чем узкоспециализированные;**
5. **Гибкость важнее, чем оптимизация;**
6. Усовершенствование в развитии важнее, чем стремление к изначальному совершенству.



# Общие сервисы важнее, чем узкоспециализированные

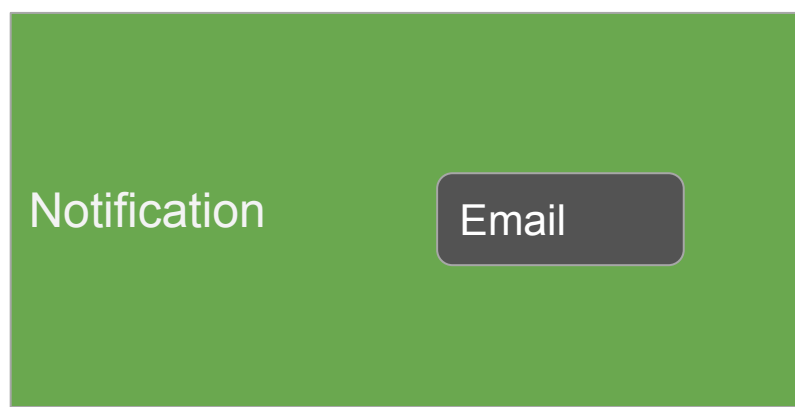


# Общие сервисы важнее, чем узкоспециализированные

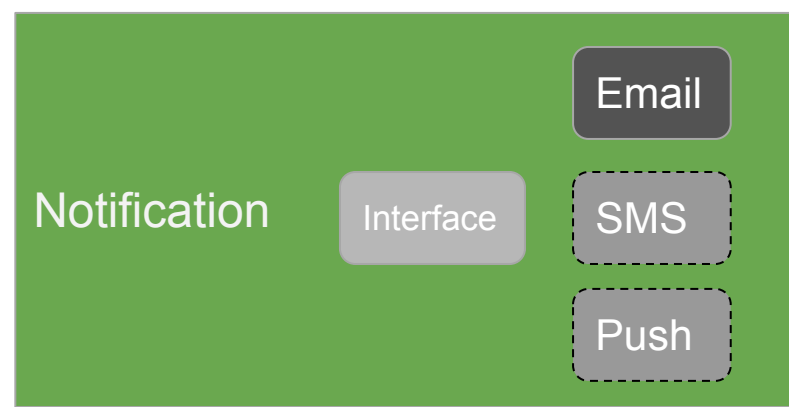


Skillbox

# Гибкость важнее, чем оптимизация



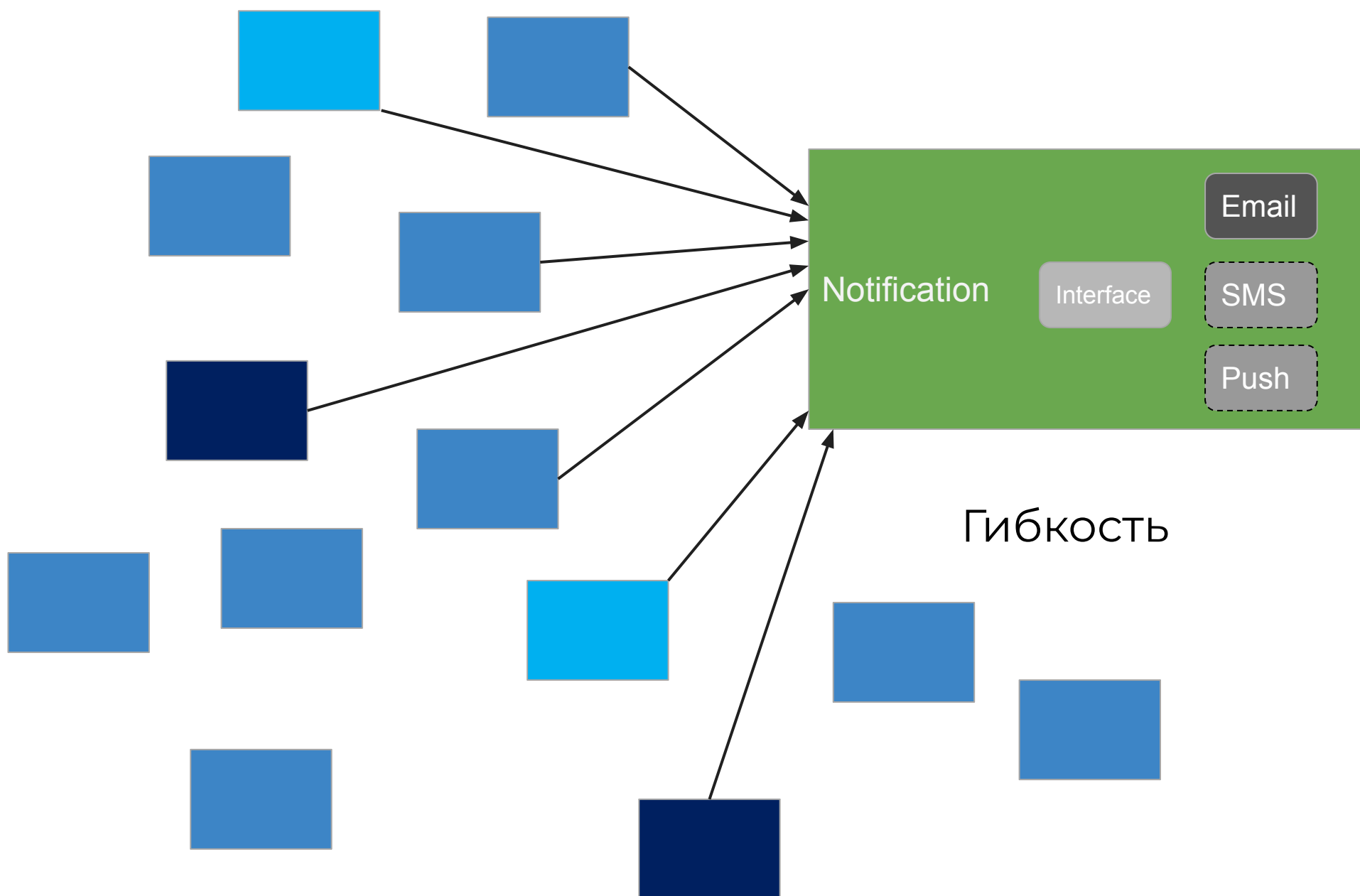
Оптимизация и  
скорость



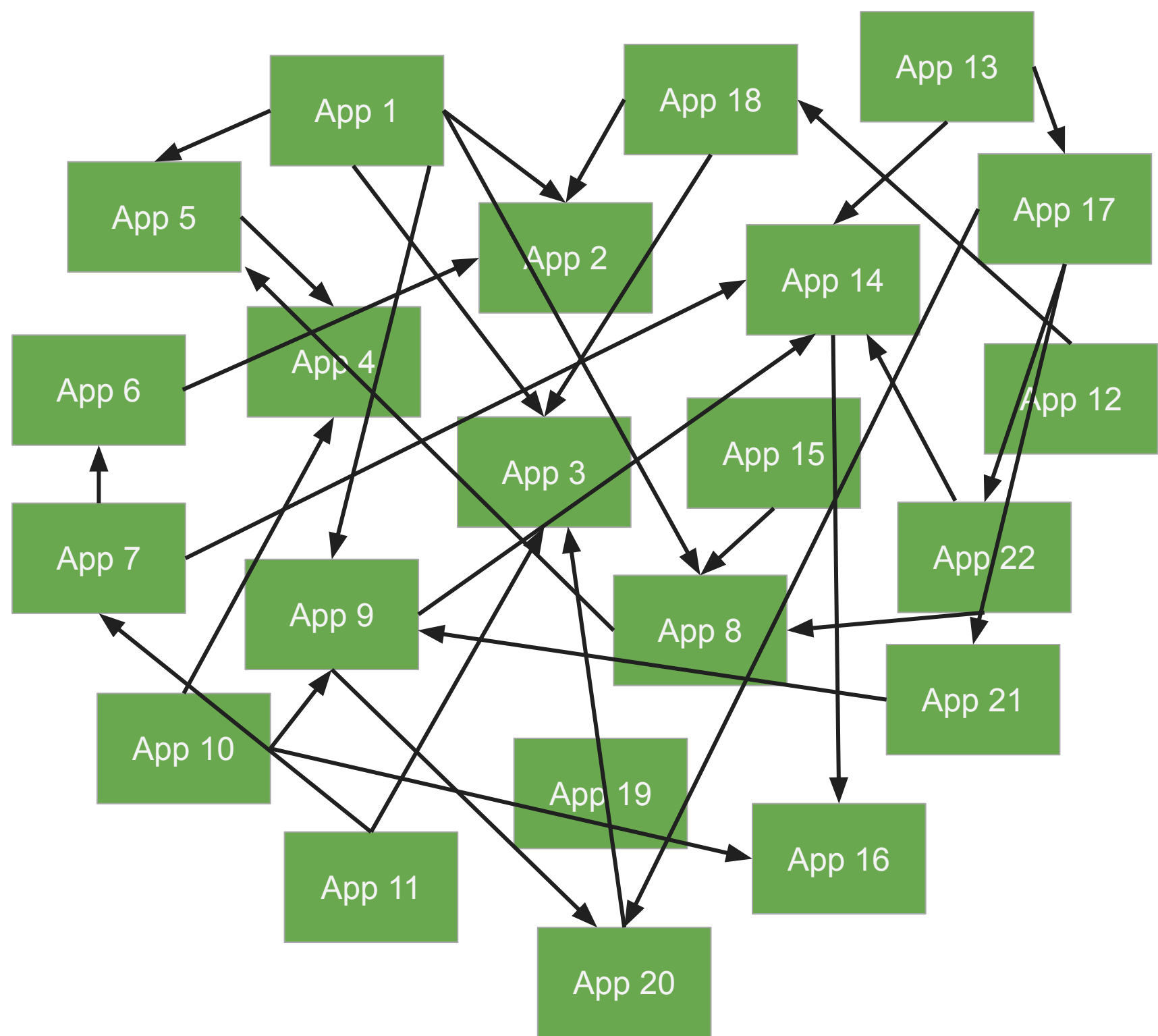
Гибкость

Skillbox

# Гибкость важнее, чем оптимизация



# SOA on Enterprise Level



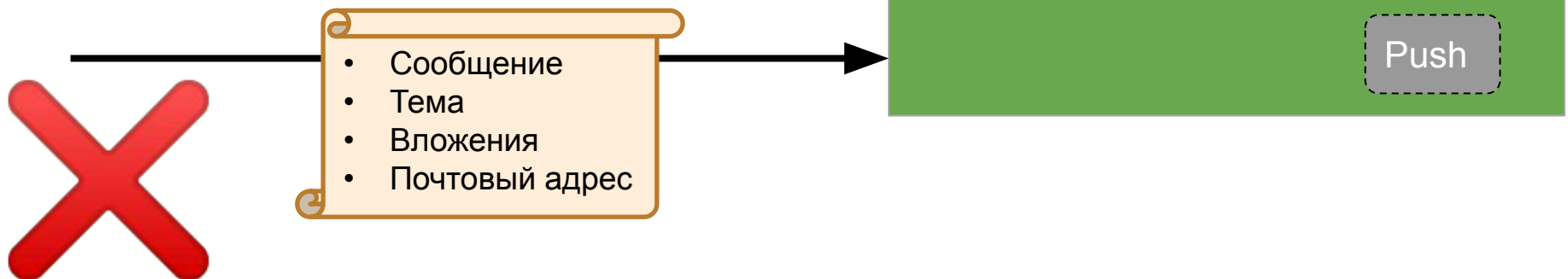
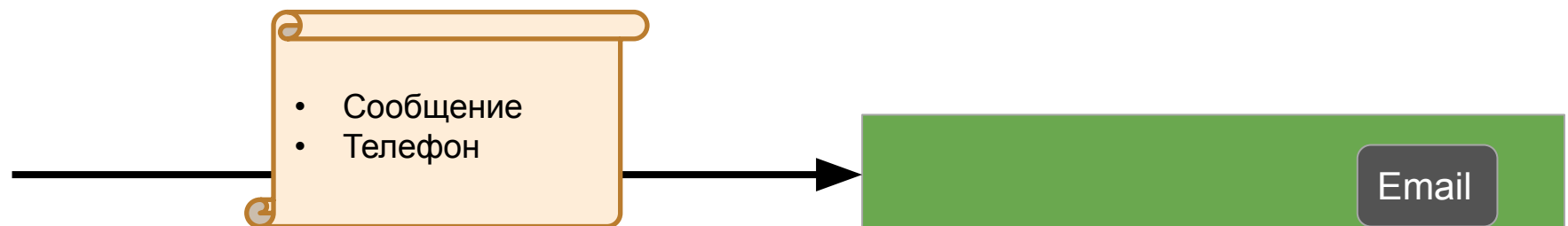
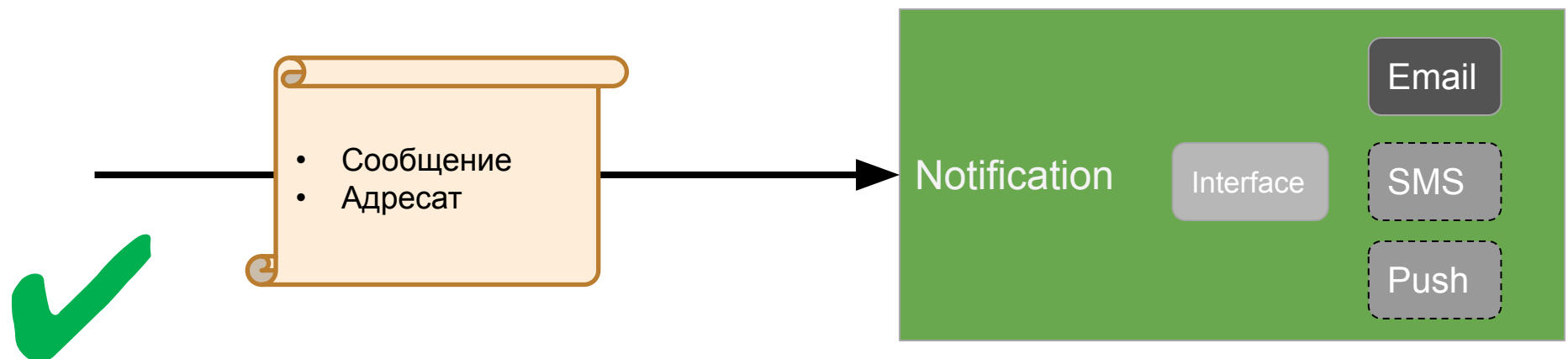
# А как надо было?



# **Не бывает хорошей или плохой архитектуры**

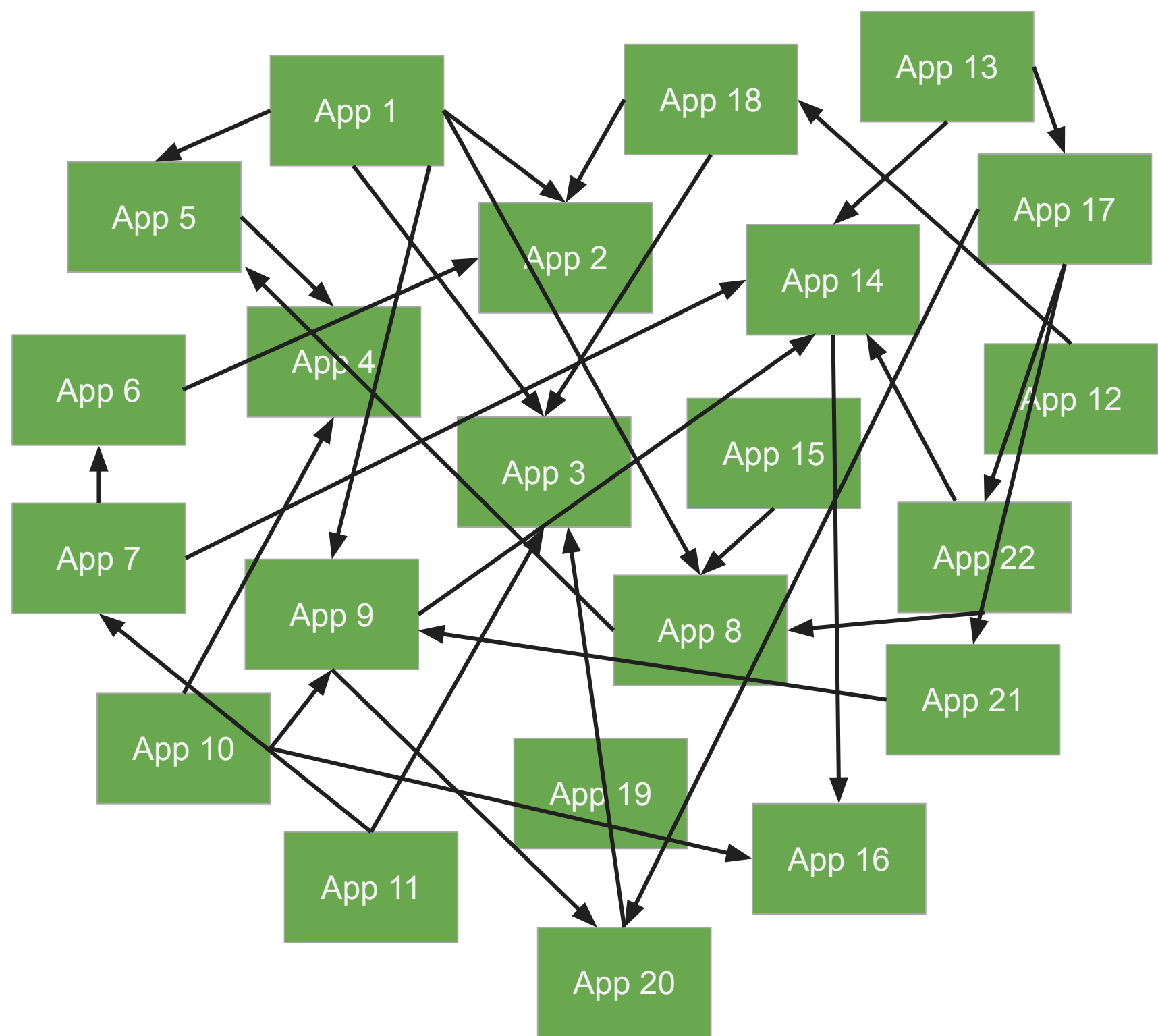
Архитектура может подходить или нет под задачи бизнеса

# It depends...

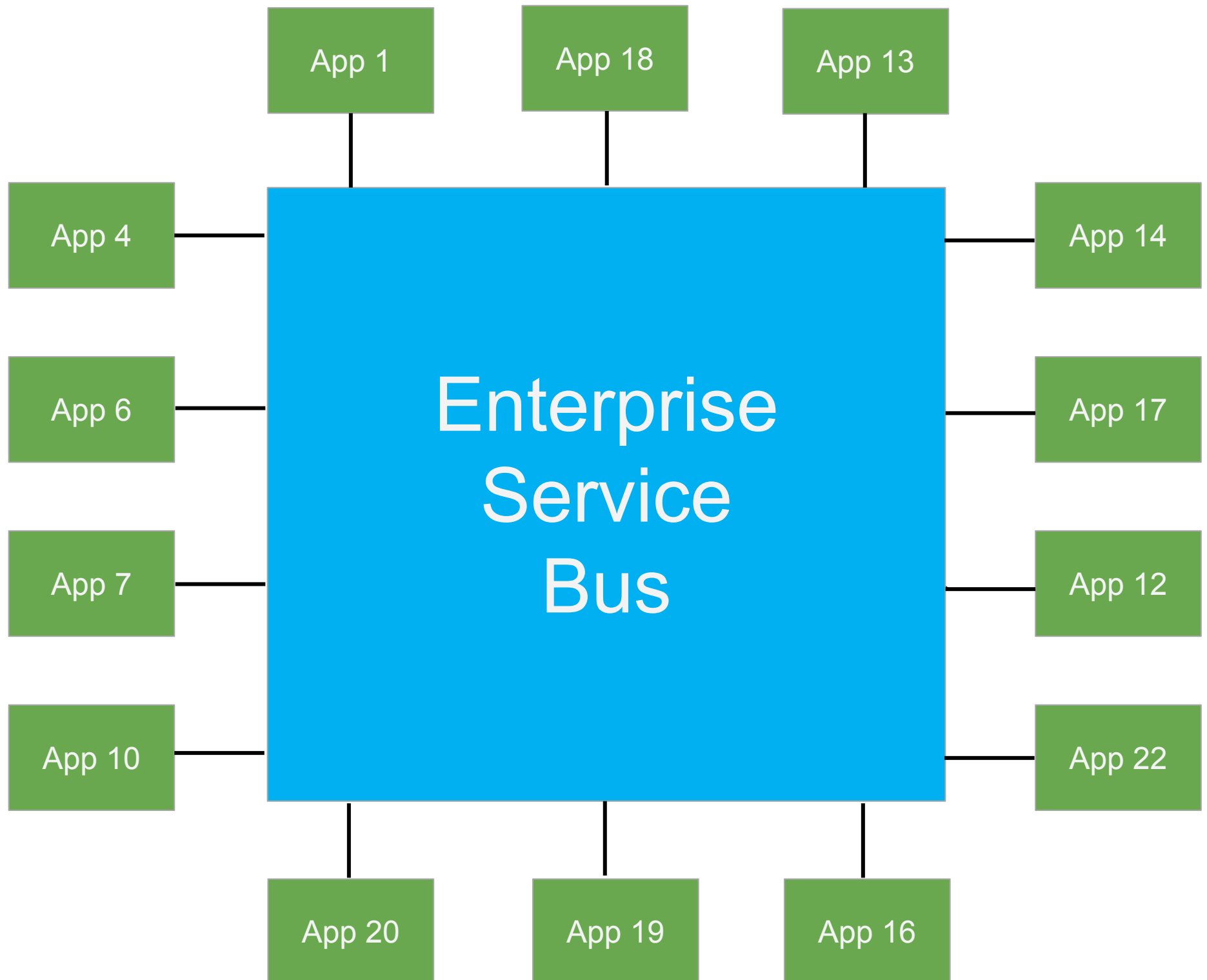




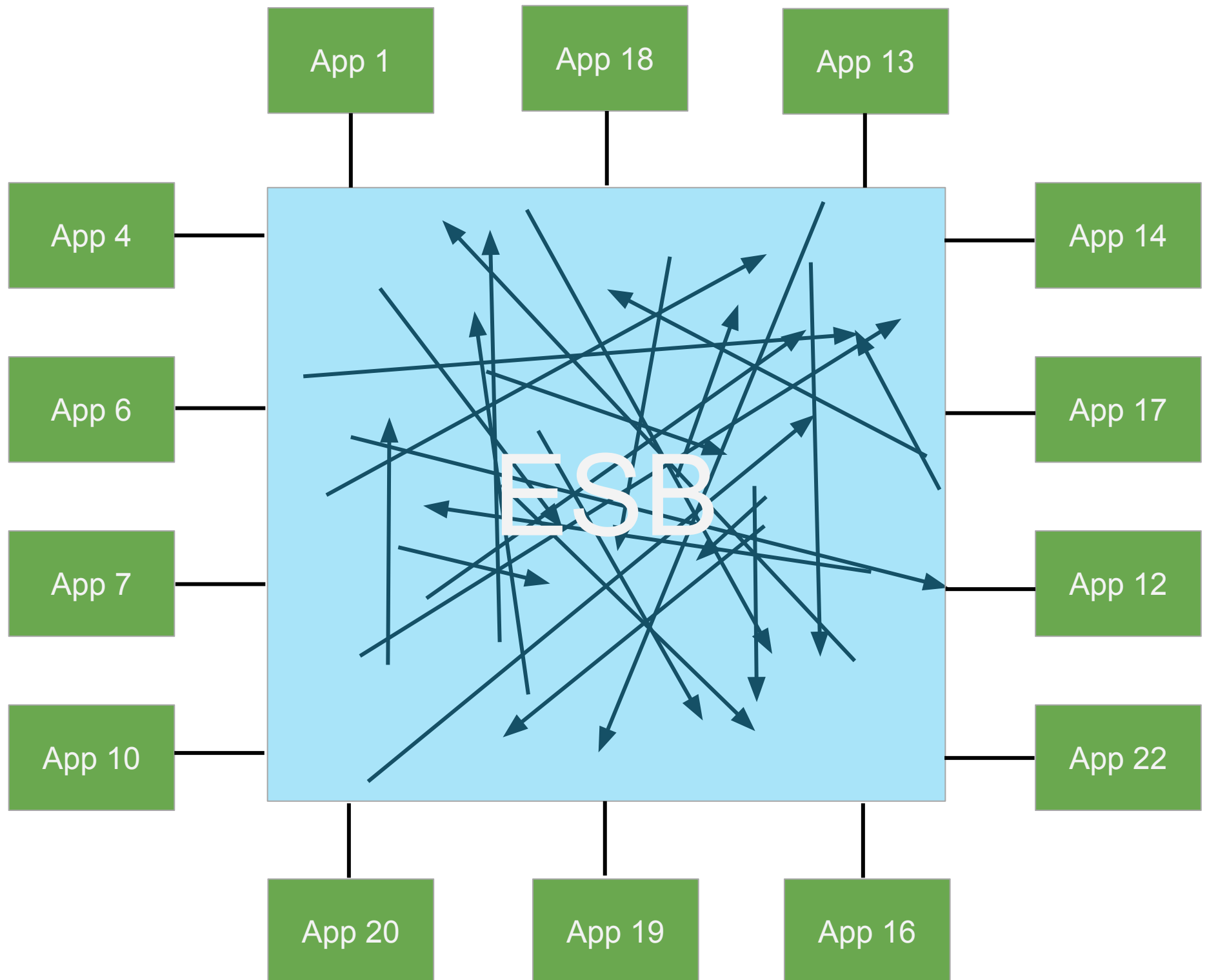
# SOA on Enterprise Level



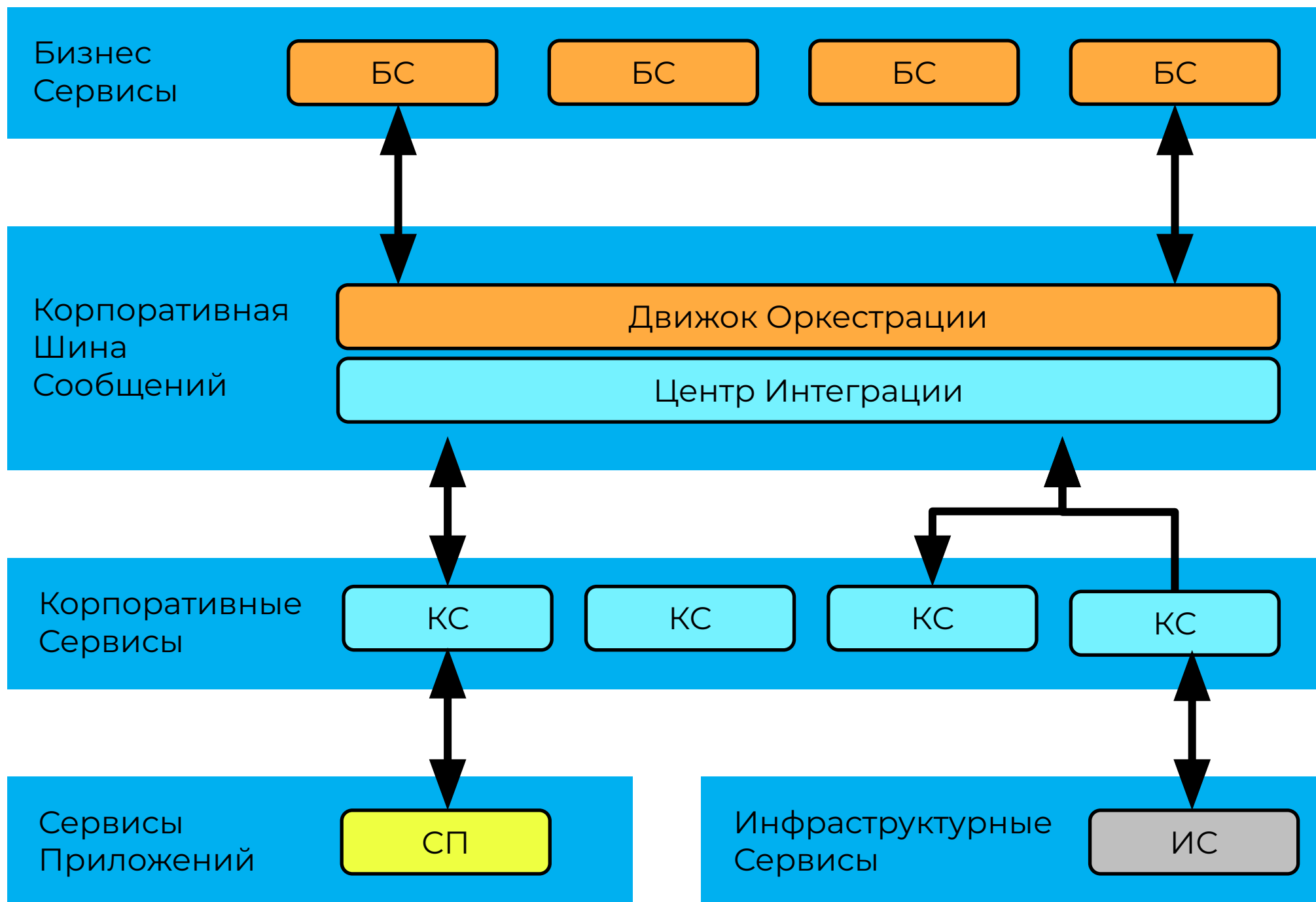
# SOA on Enterprise Level



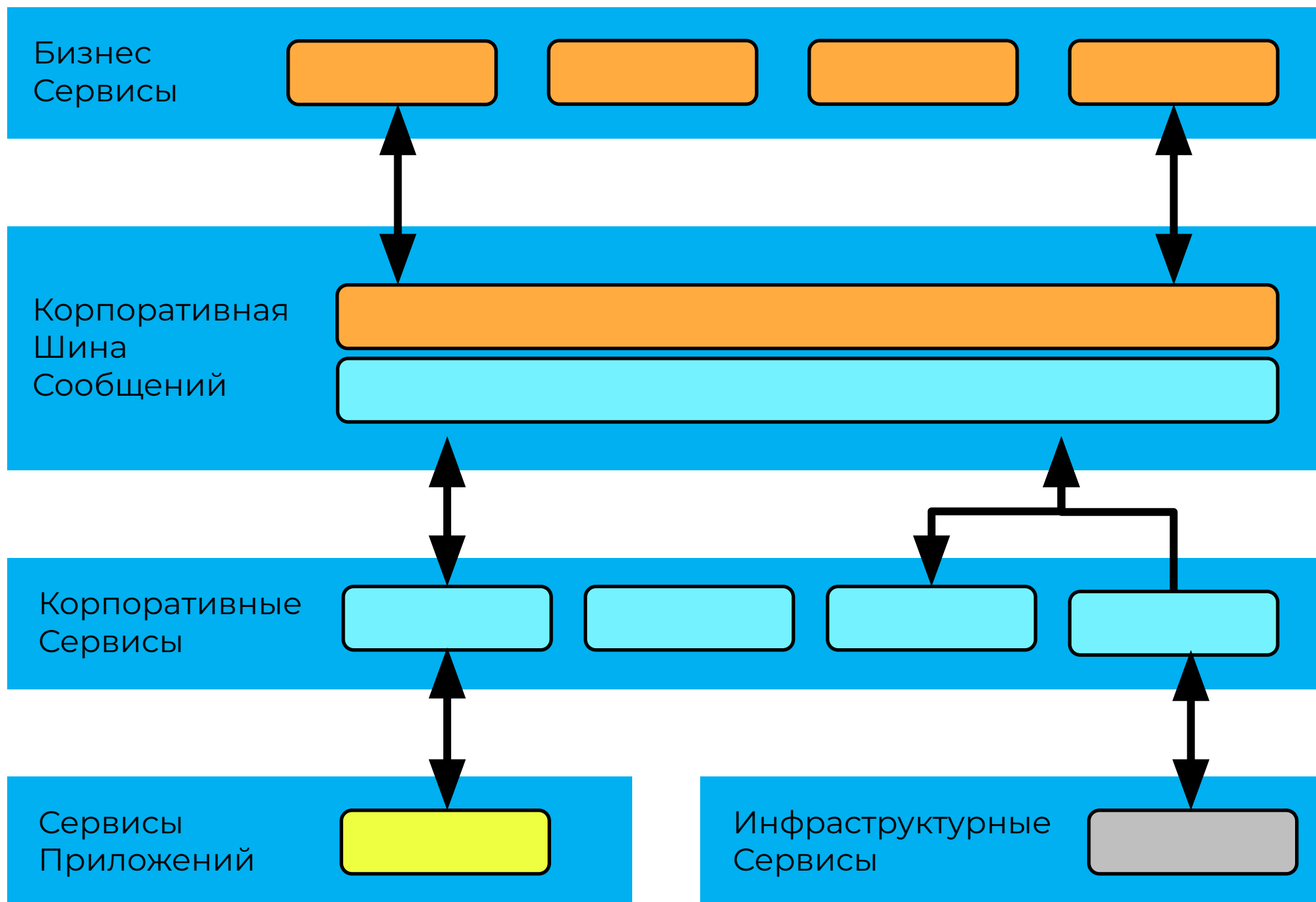
# SOA on Enterprise Level



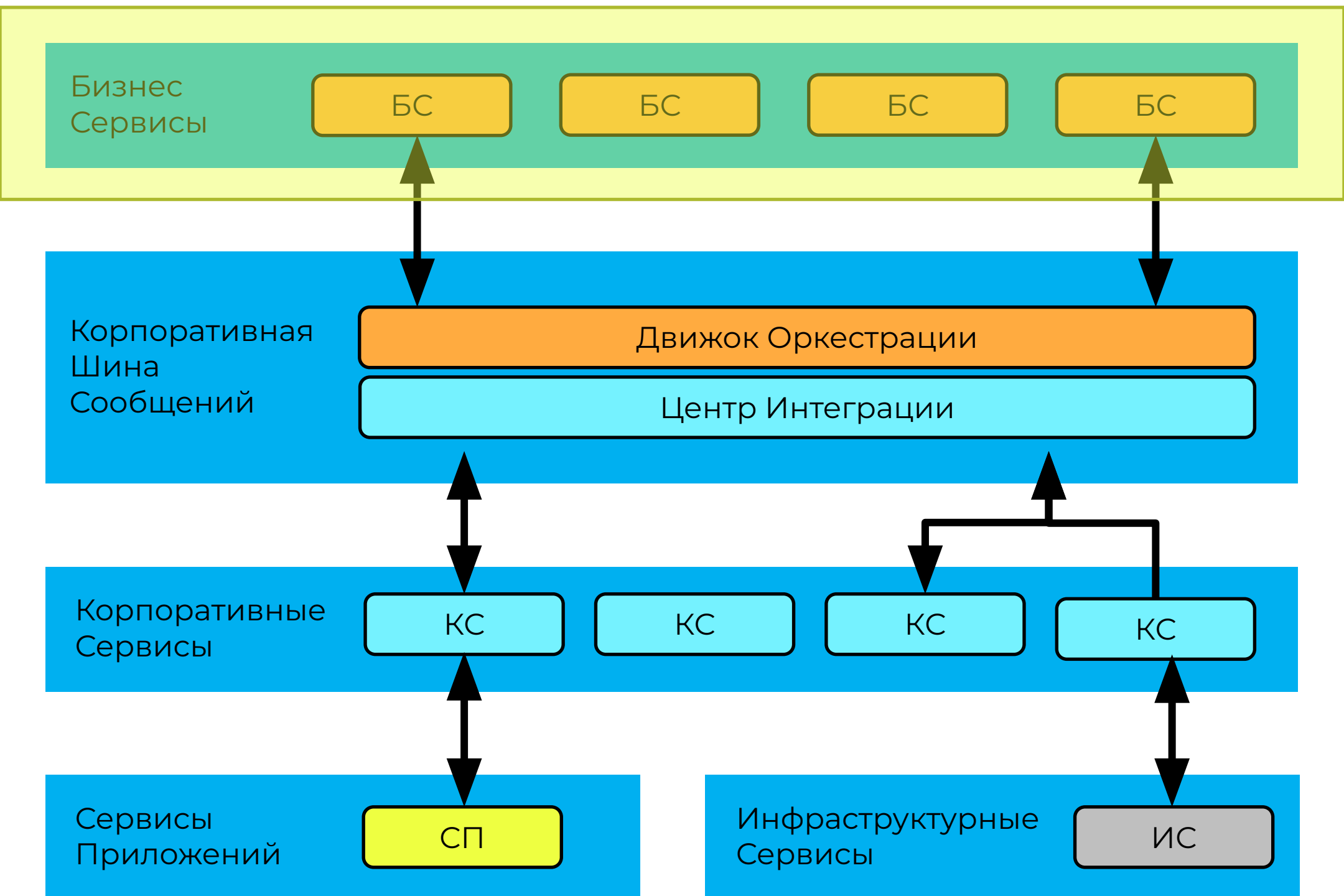
# Топология



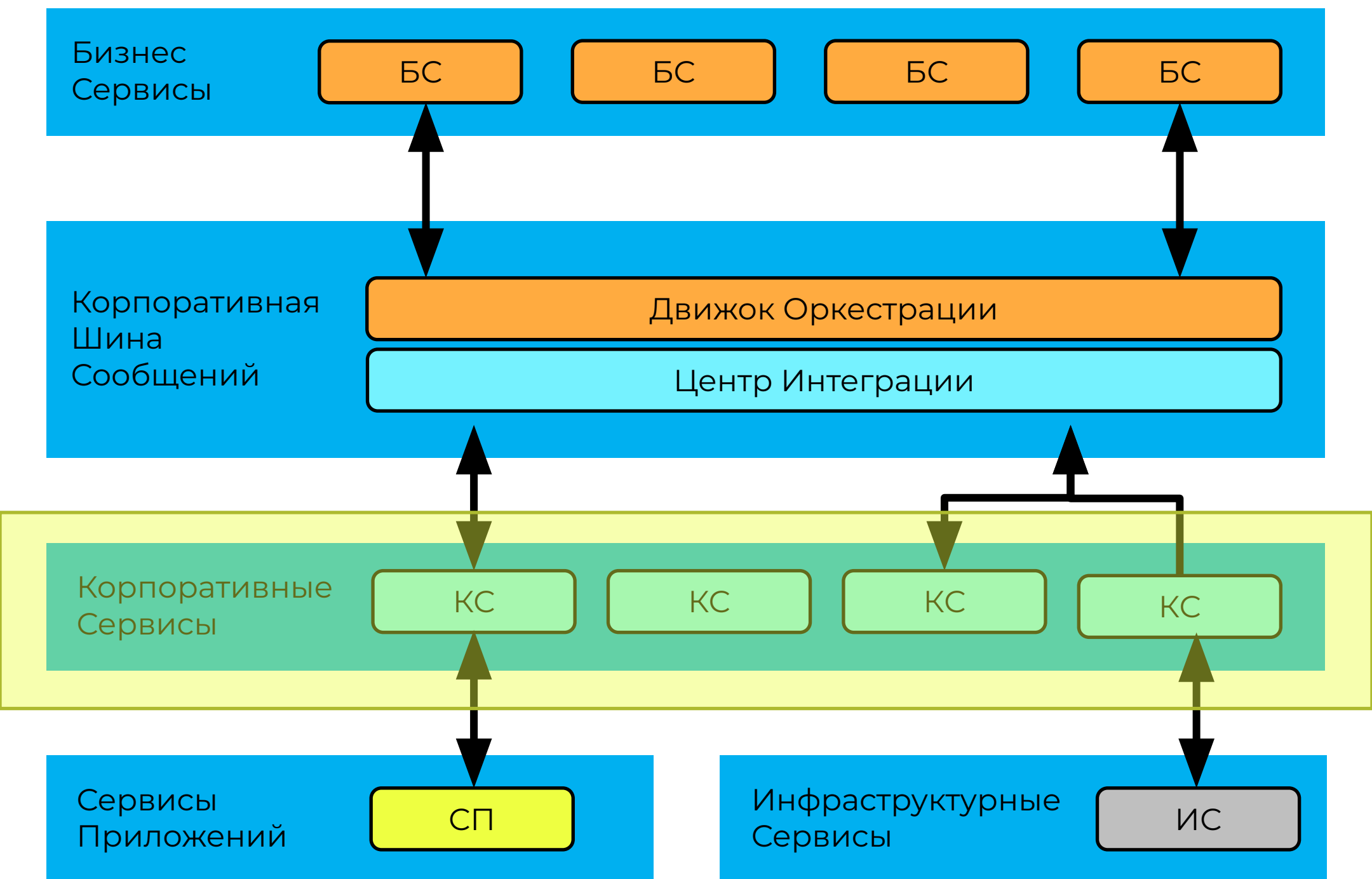
# Топология



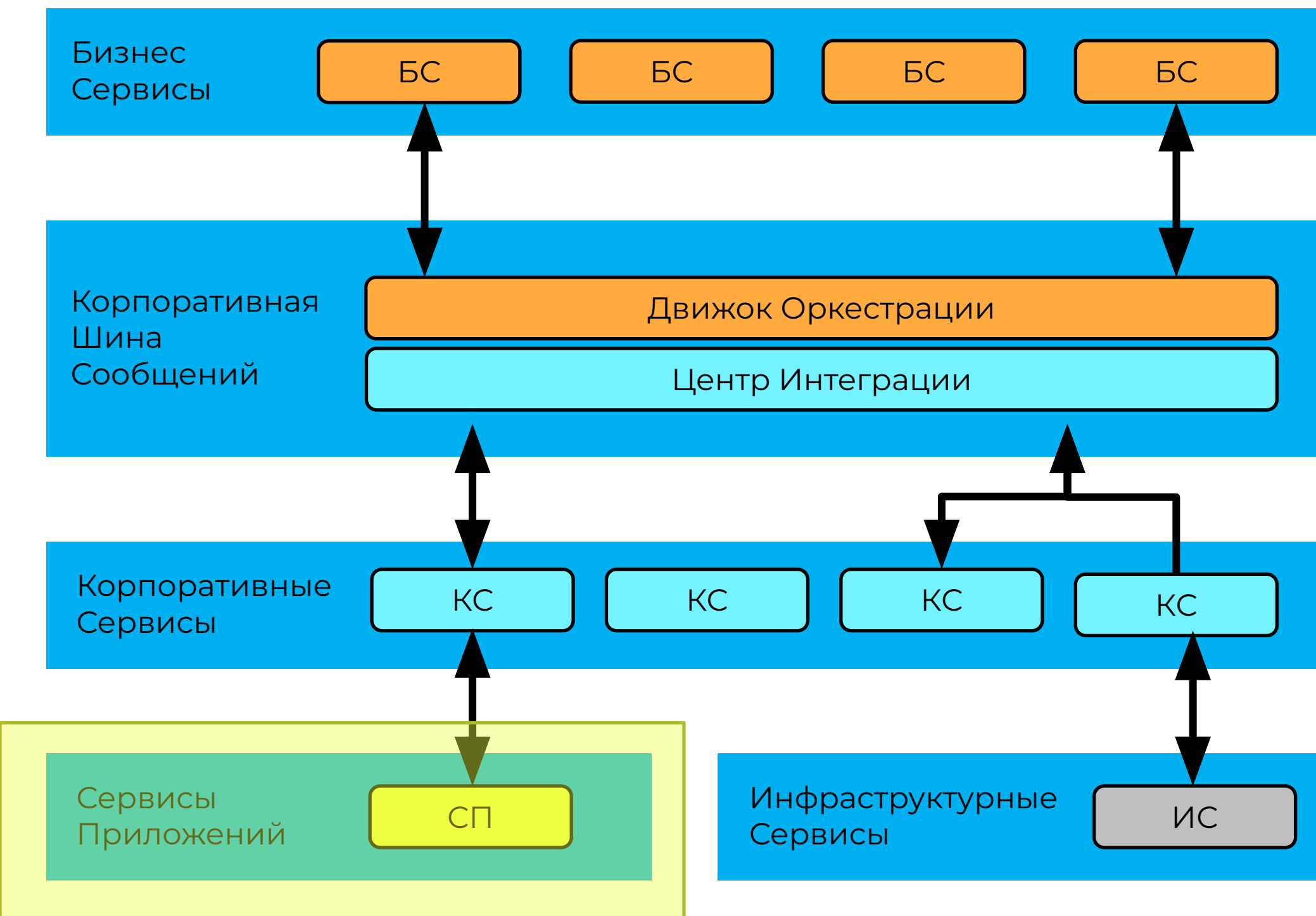
# Топология



# Топология

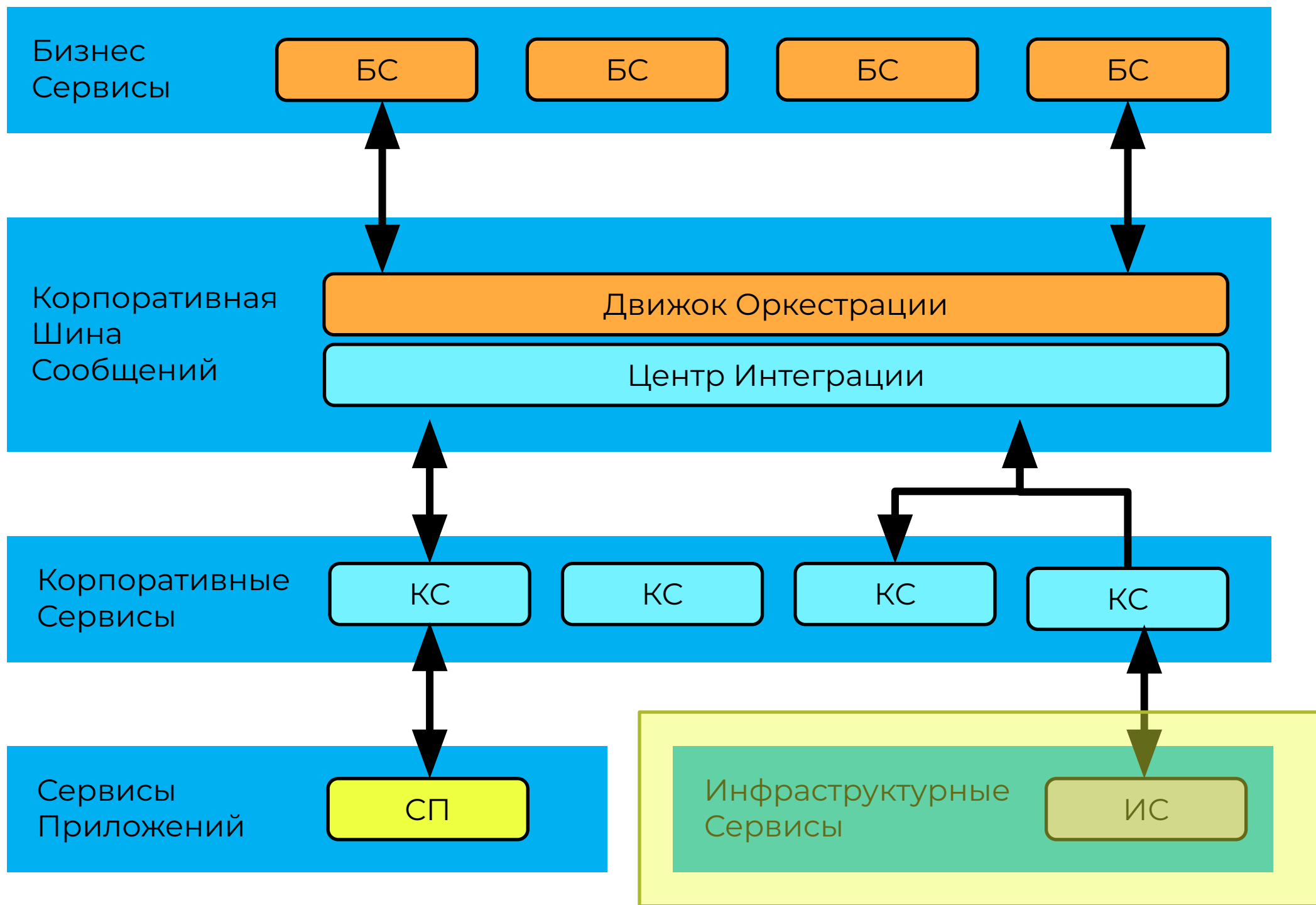


# Топология

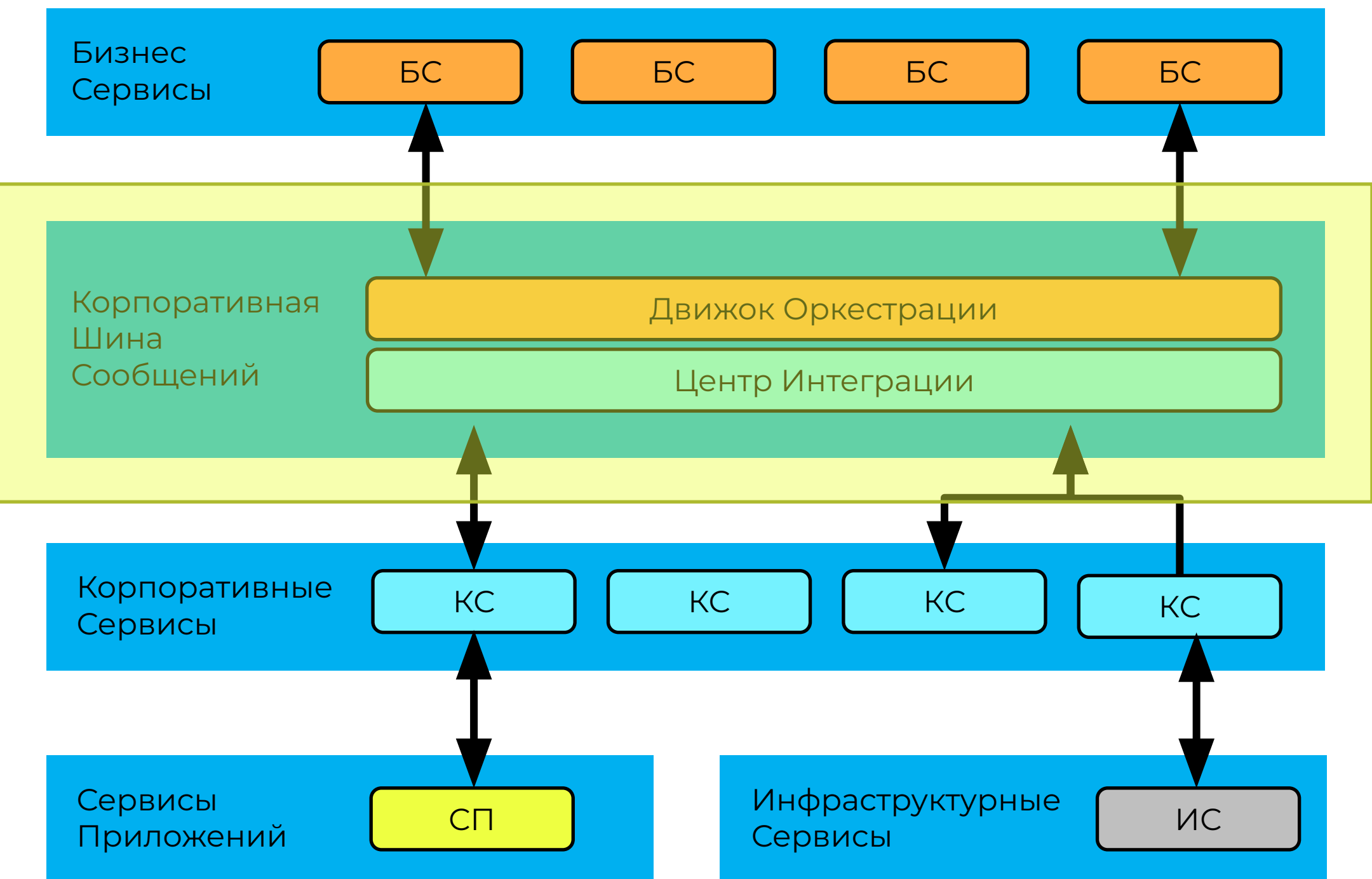




# Топология



# Топология



# Выводы

## Плюсы:

- Отличный потенциал для общения разнородных систем
- Категоризация и таксономия бизнес-сервисов
- Эластичность общей системы в теории

## Минусы:

- Очень сложное разворачивание новых версий и тестируемость
- Скорость работы
- Сложность интеграции
- Цена покупки и владения.

# Самое важное

Распределенные транзакции очень сложны с точки зрения бизнеса и технологии.

# Что дальше?

- Как Service-Oriented стиль эволюционировал в Service-Based
- Принципы Service-Based Architecture