

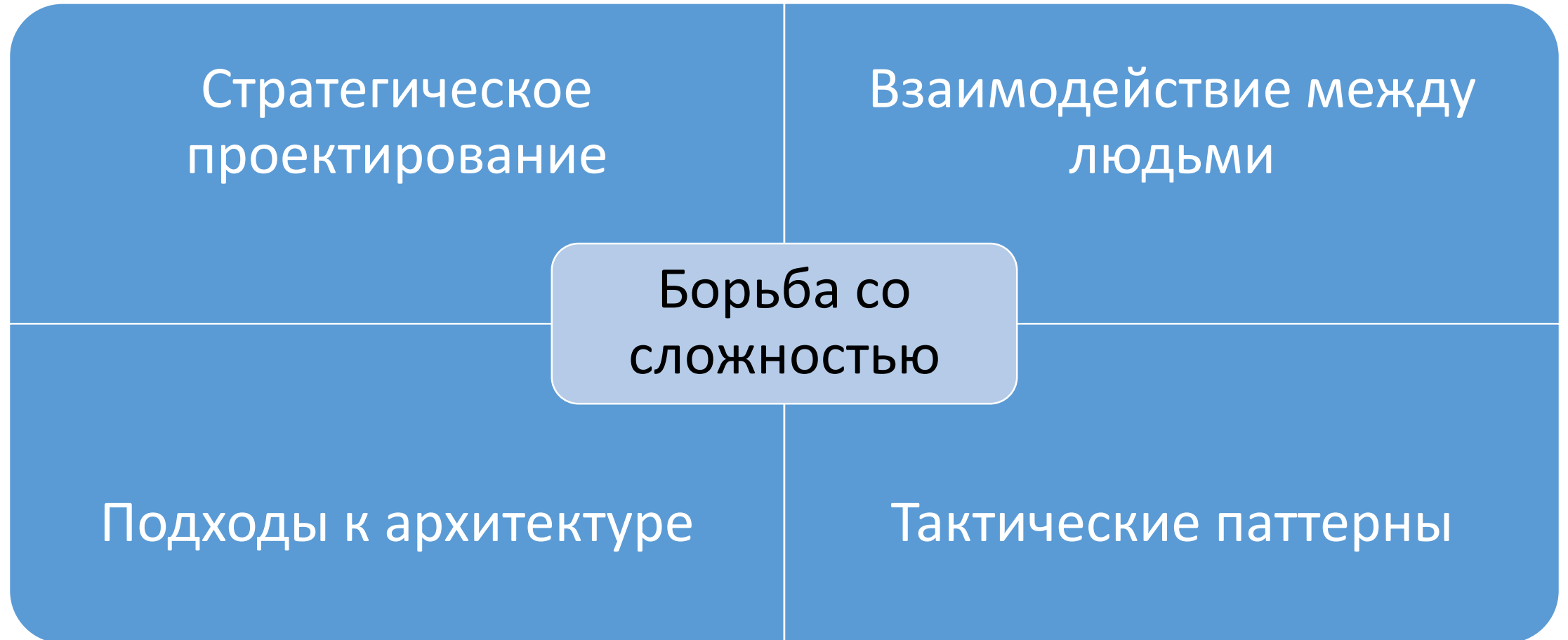
# Domain Driven Design: профит малой кровью

Алексей Мерсон  
HiQo Solutions

“There are few things that are less  
logical than business logic”

Martin Fowler

# Domain Driven Design:



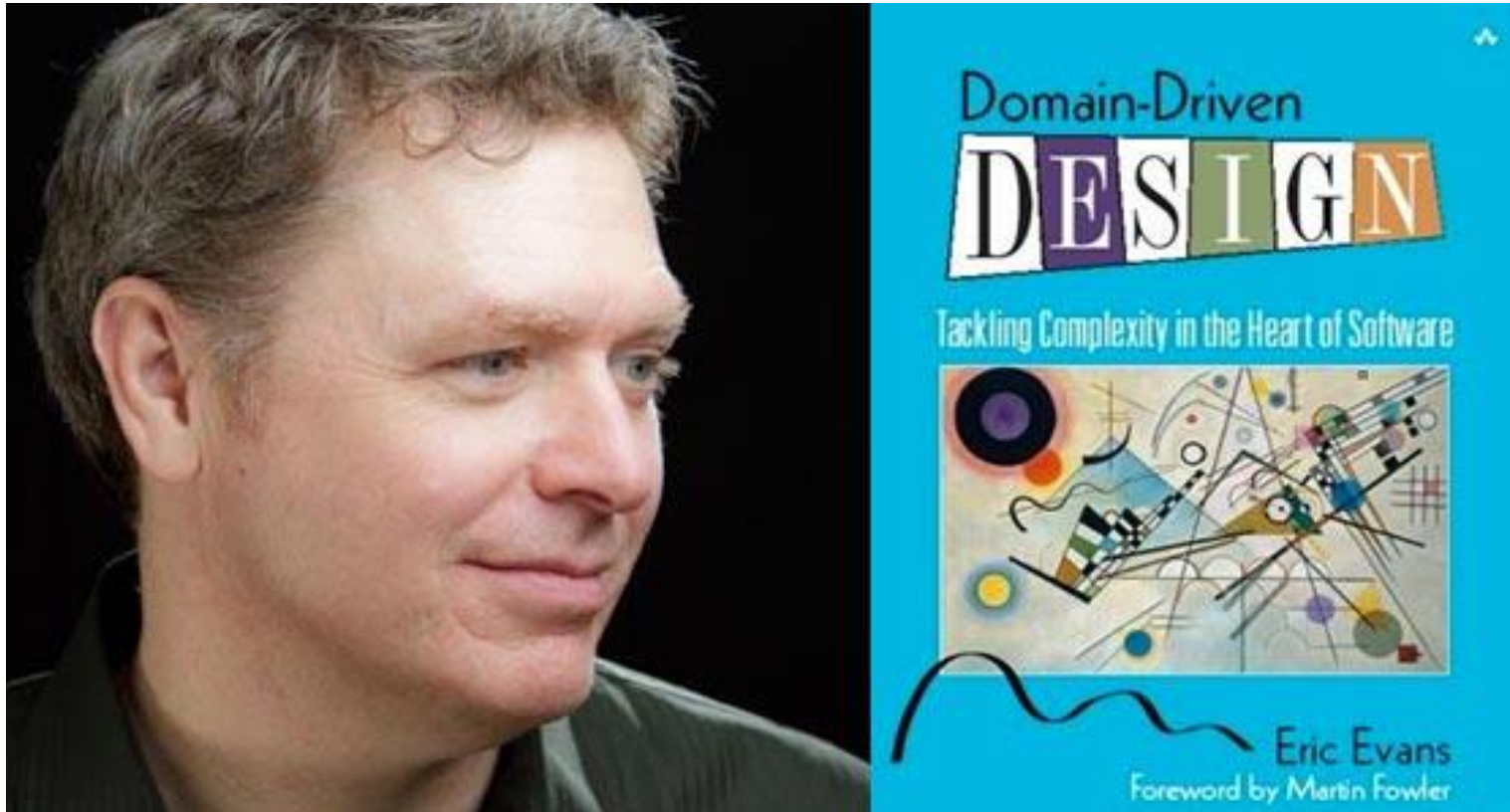
# Domain Driven Design:

- Автор термина Эрик Эванс



# Domain Driven Design:

- Автор термина Эрик Эванс      2003 год



# Стратегическое и тактическое проектирование

Repository  
Entity  
Aggregate  
Factory  
Module  
Domain  
Value object  
Domain service  
Domain event  
Separated Interface

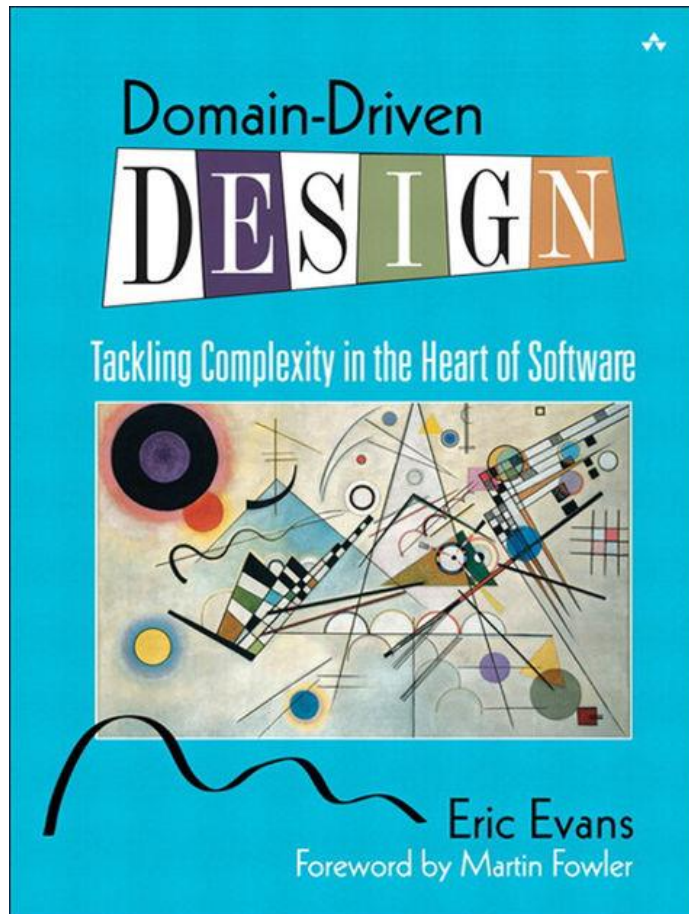
# Стратегическое и тактическое проектирование

Context Map  
Bounded Context  
Anticorruption Layer  
Ubiquitous Language  
Domain model  
Subdomains

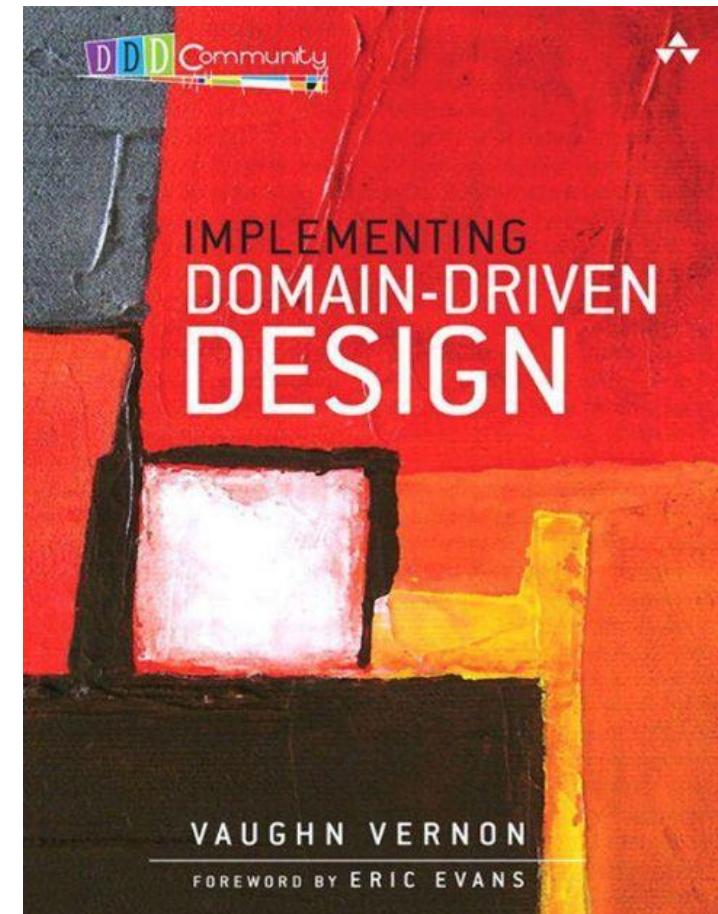
Repository  
Entity  
Aggregate  
Factory  
Module  
Domain event  
Separated Interface  
Value object  
Domain service object

# Domain Driven Design:

Blue book (2003)



Red book (2013)





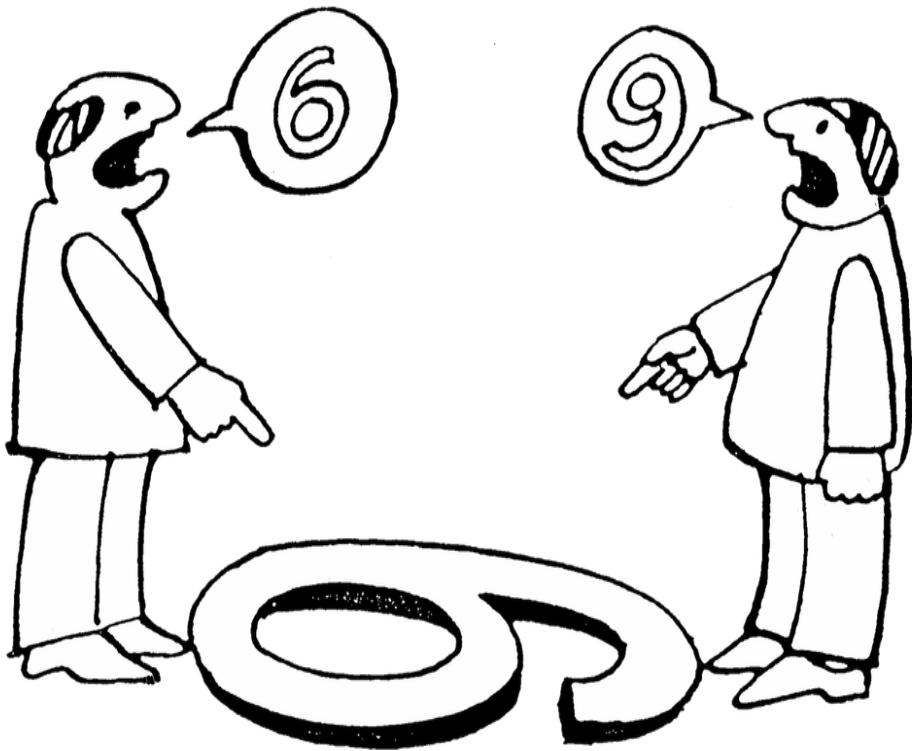
# Ключевые идеи стратегического проектирования

# Ключевые идеи стратегического проектирования



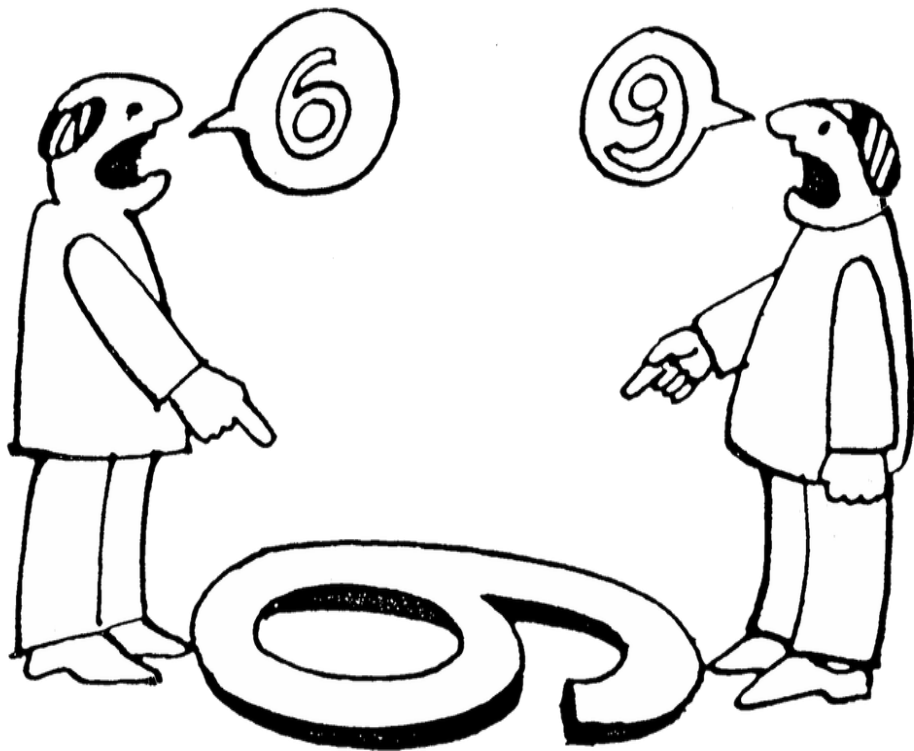
# Ключевые идеи стратегического проектирования

Доменные эксперты

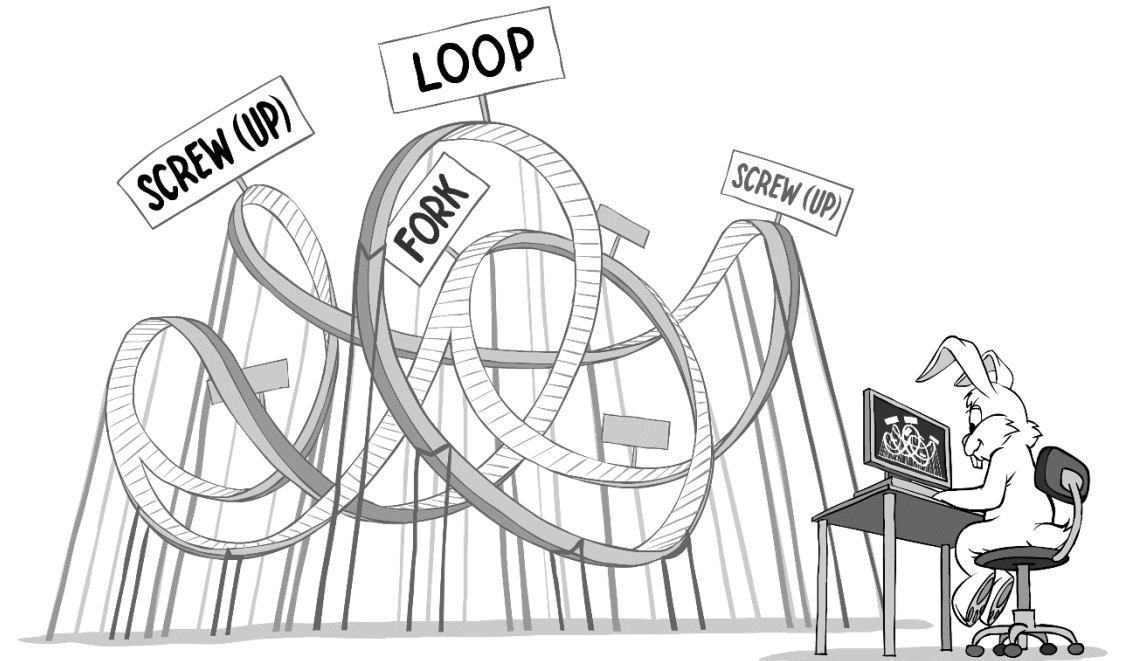


# Ключевые идеи стратегического проектирования

Доменные эксперты



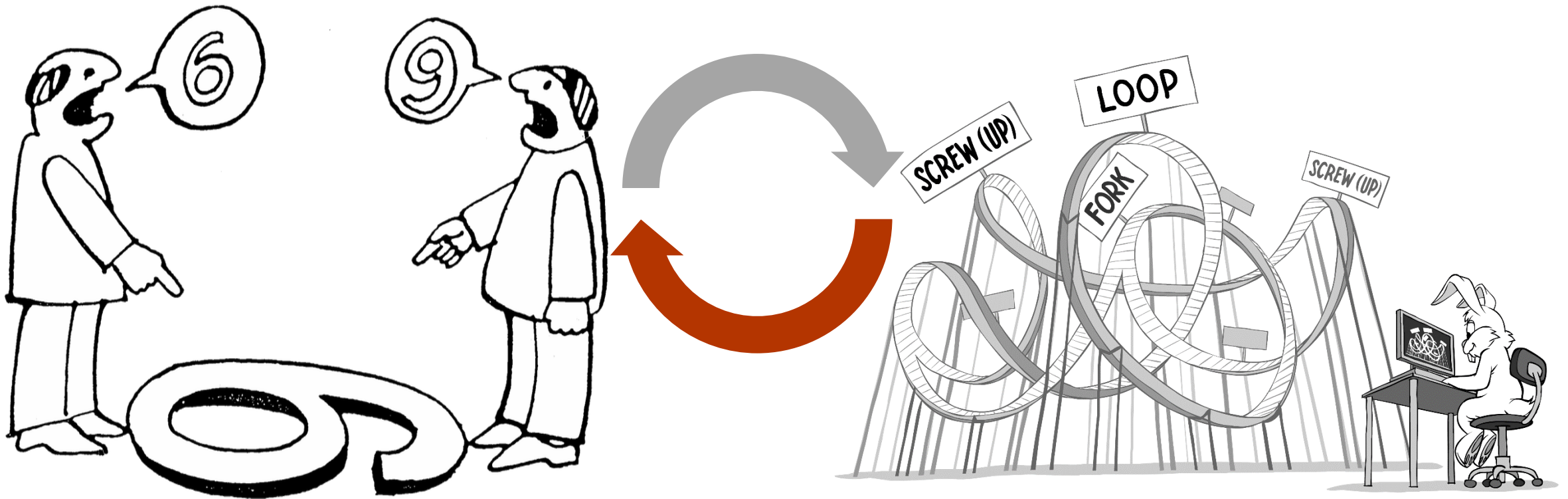
Технические специалисты



# Ключевые идеи стратегического проектирования

Доменные эксперты

Технические специалисты

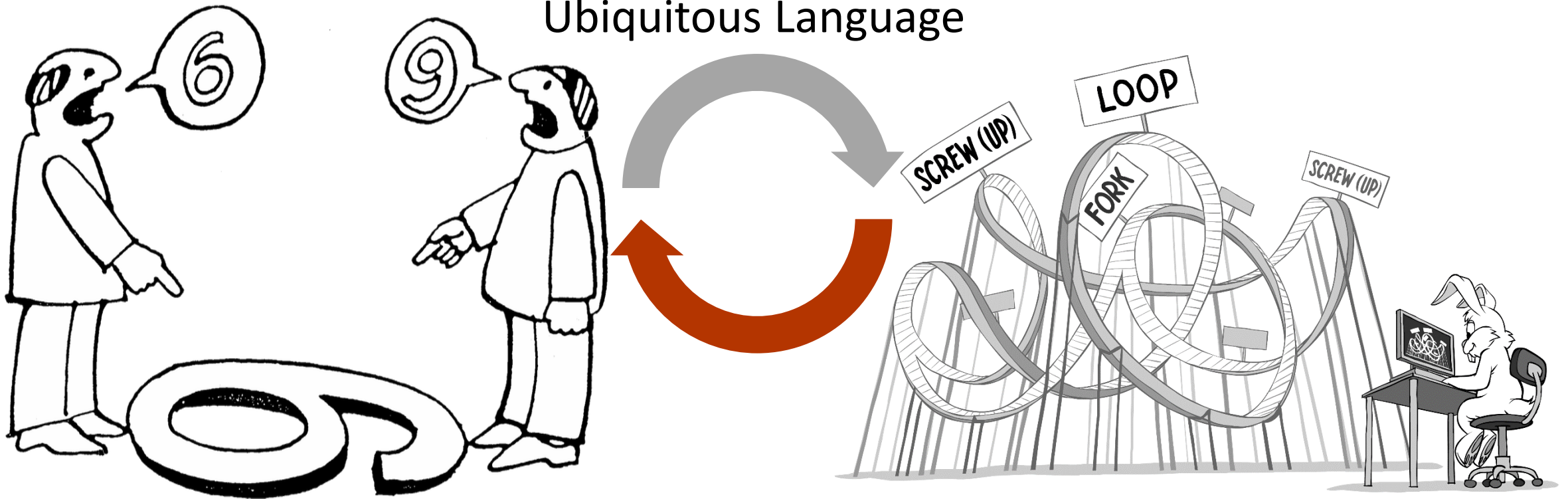


# Ключевые идеи стратегического проектирования

Доменные эксперты

Технические специалисты

Ubiquitous Language



# Ключевые идеи стратегического проектирования

1. Общение между участниками проекта формирует ubiquitous language

# Ключевые идеи стратегического проектирования

Проект для CompanyName



# Ключевые идеи стратегического проектирования

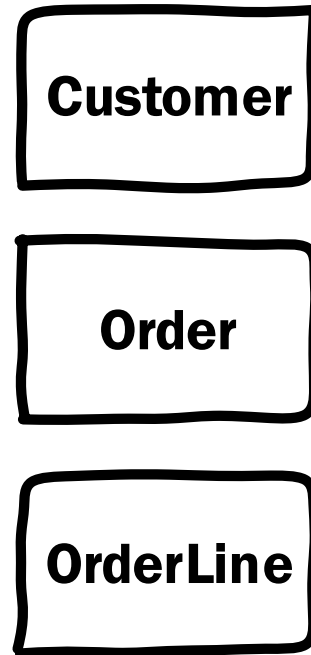
Проект для CompanyName:  
интернет-магазин BestShop

# Ключевые идеи стратегического проектирования

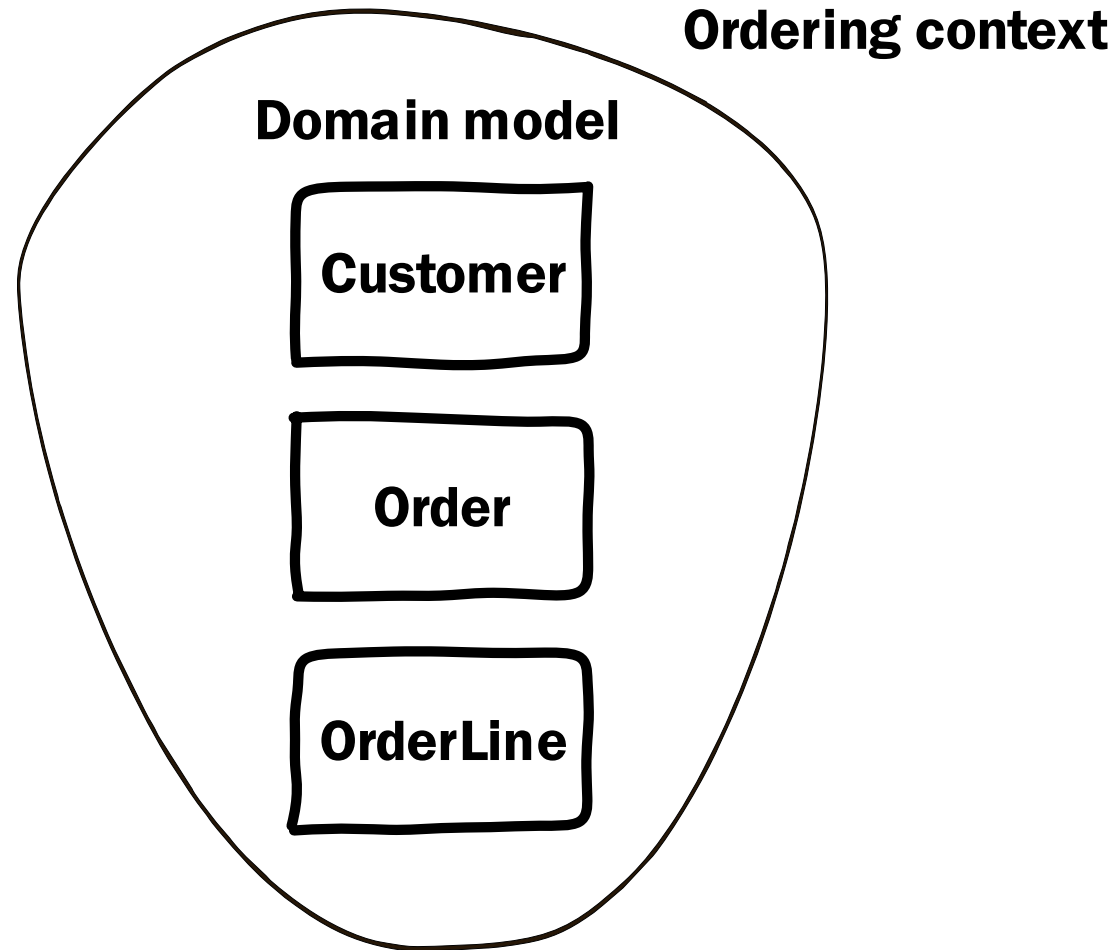
Изучаем бизнес-требования

# Ключевые идеи стратегического проектирования

## Domain model



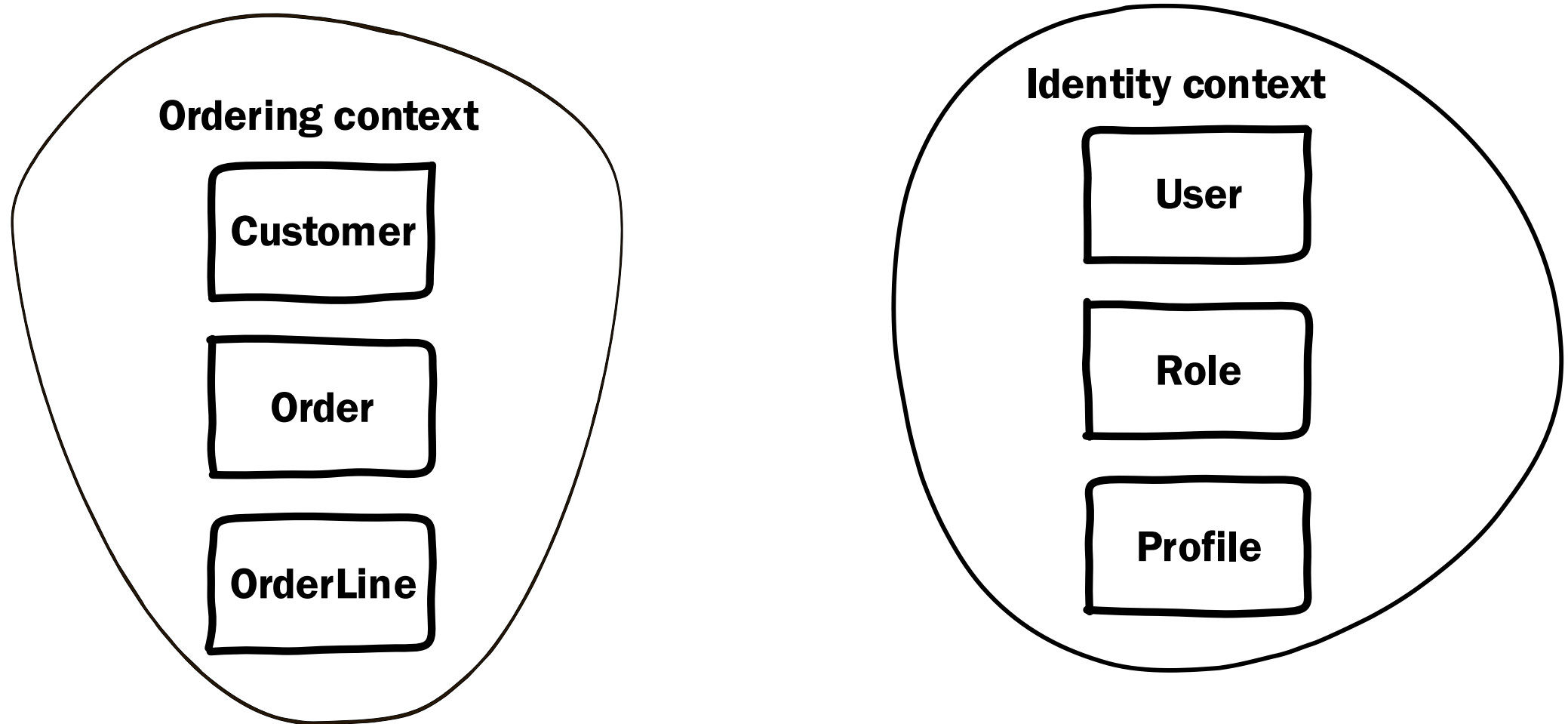
# Ключевые идеи стратегического проектирования



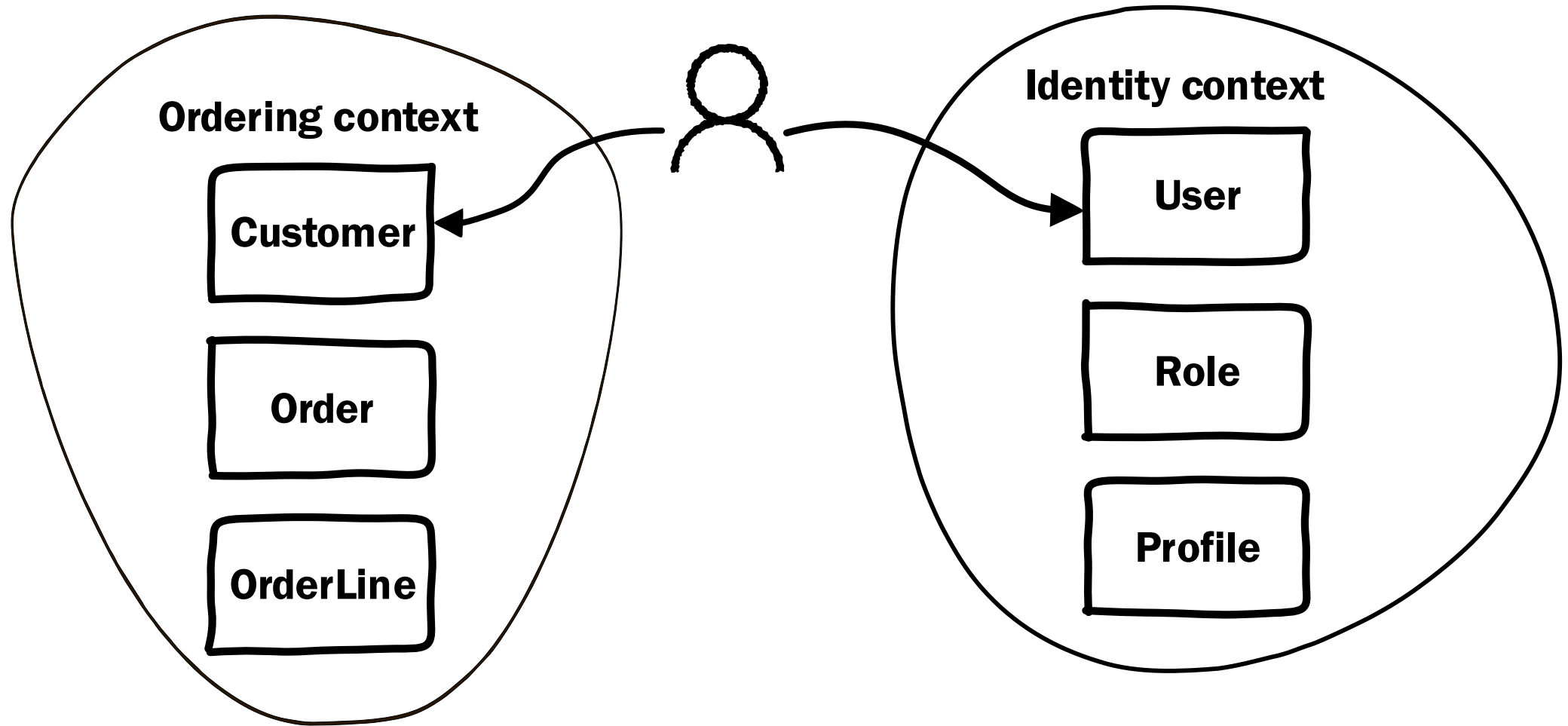
# Ключевые идеи стратегического проектирования

1. Общение между участниками проекта формирует ubiquitous language
2. Доменная модель и ubiquitous language ограничены контекстом (bounded context)

# Ключевые идеи стратегического проектирования



# Ключевые идеи стратегического проектирования



# Ключевые идеи стратегического проектирования

1. Общение между участниками проекта формирует ubiquitous language
2. Доменная модель и ubiquitous language ограничены контекстом (bounded context)



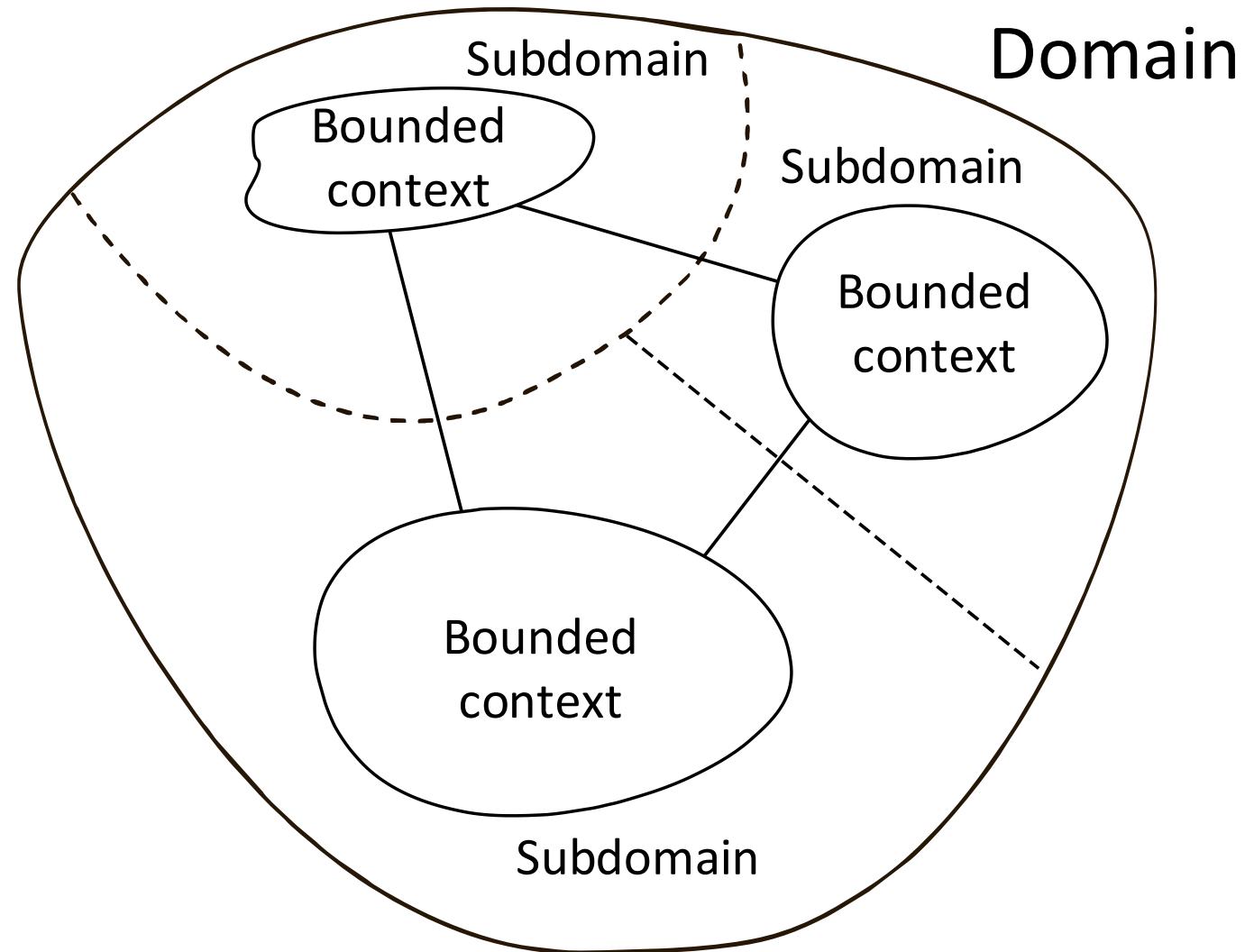
# Ключевые идеи стратегического проектирования

Ordering microservice

# Ключевые идеи стратегического проектирования

1. Общение между участниками проекта формирует ubiquitous language
2. Доменная модель и ubiquitous language ограничены контекстом (bounded context)
3. Доменная модель и бизнес-логика использует ubiquitous language

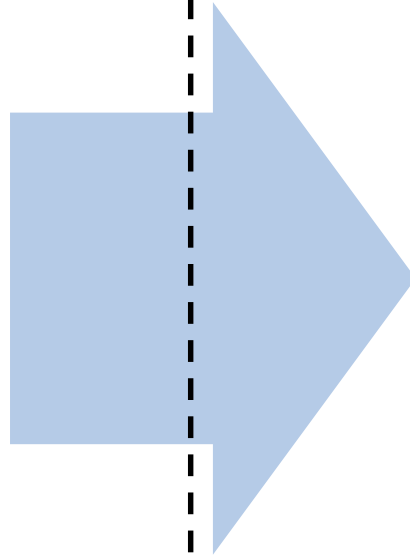
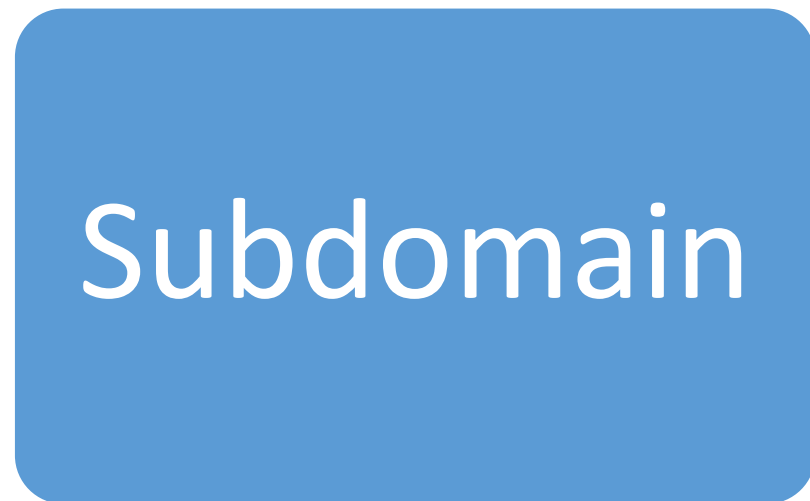
# Ключевые идеи стратегического проектирования



# Ключевые идеи стратегического проектирования

Задачи бизнеса

Программные решения



# Ключевые идеи стратегического проектирования

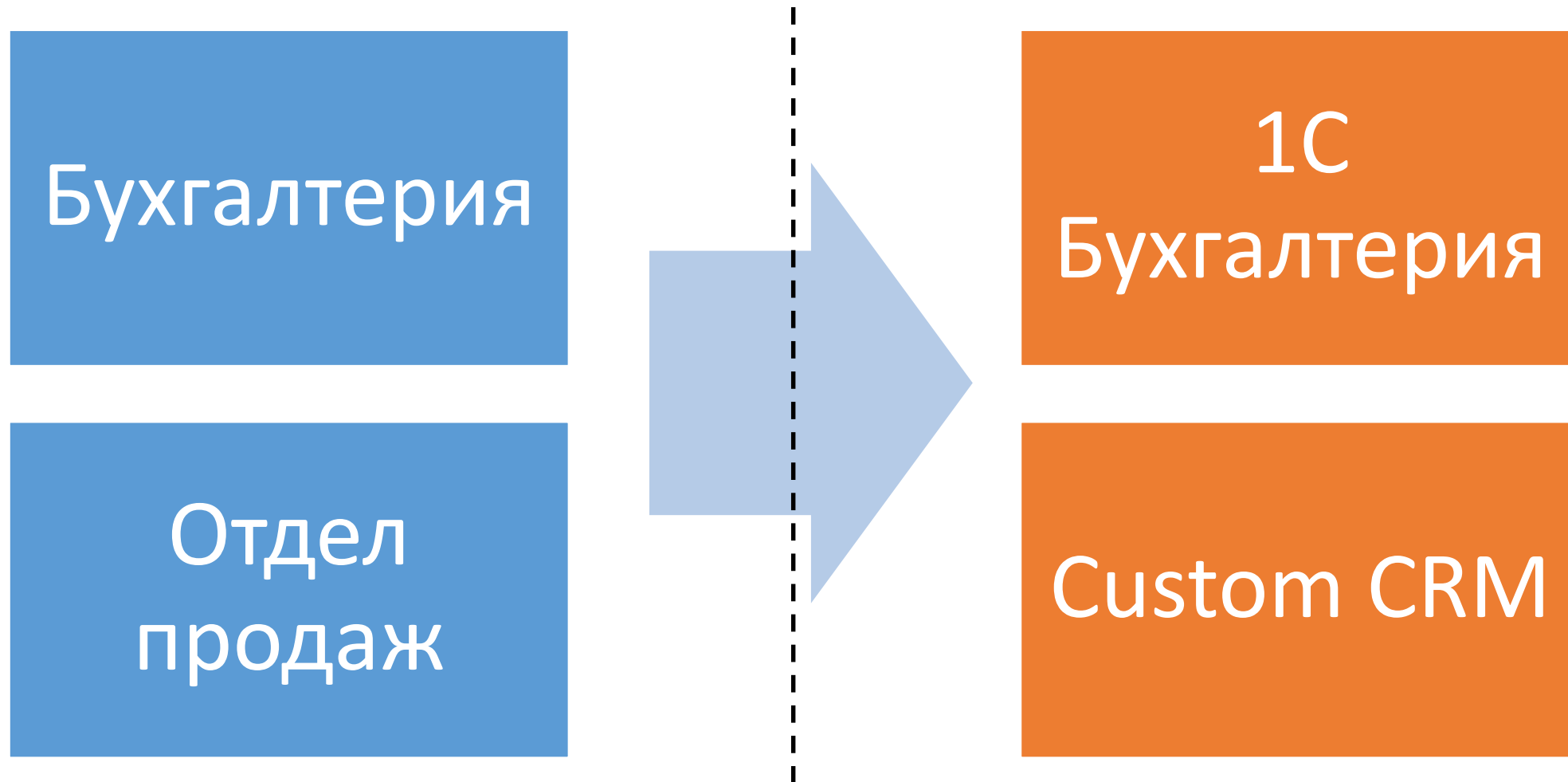
Бухгалтерия



Программные решения



# Ключевые идеи стратегического проектирования

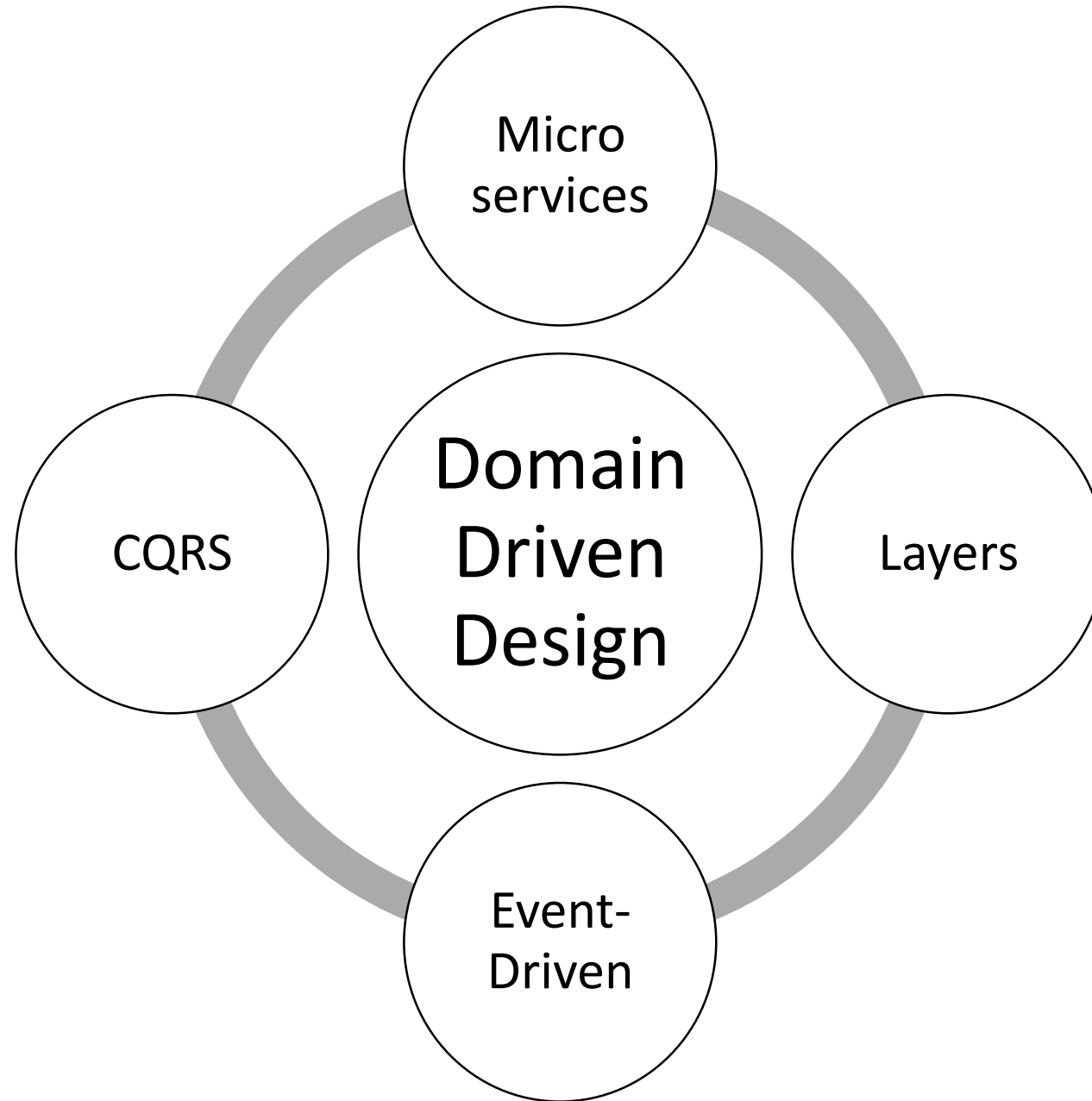


# Ключевые идеи стратегического проектирования

1. Общение между участниками проекта формирует ubiquitous language
2. Доменная модель и ubiquitous language ограничены контекстом (bounded context)
3. Доменная модель и бизнес-логика использует ubiquitous language
4. Бизнес-процессы делятся на поддомены, ПО – на контексты
5. В идеале один поддомен = один контекст

# Архитектура и управление зависимостями



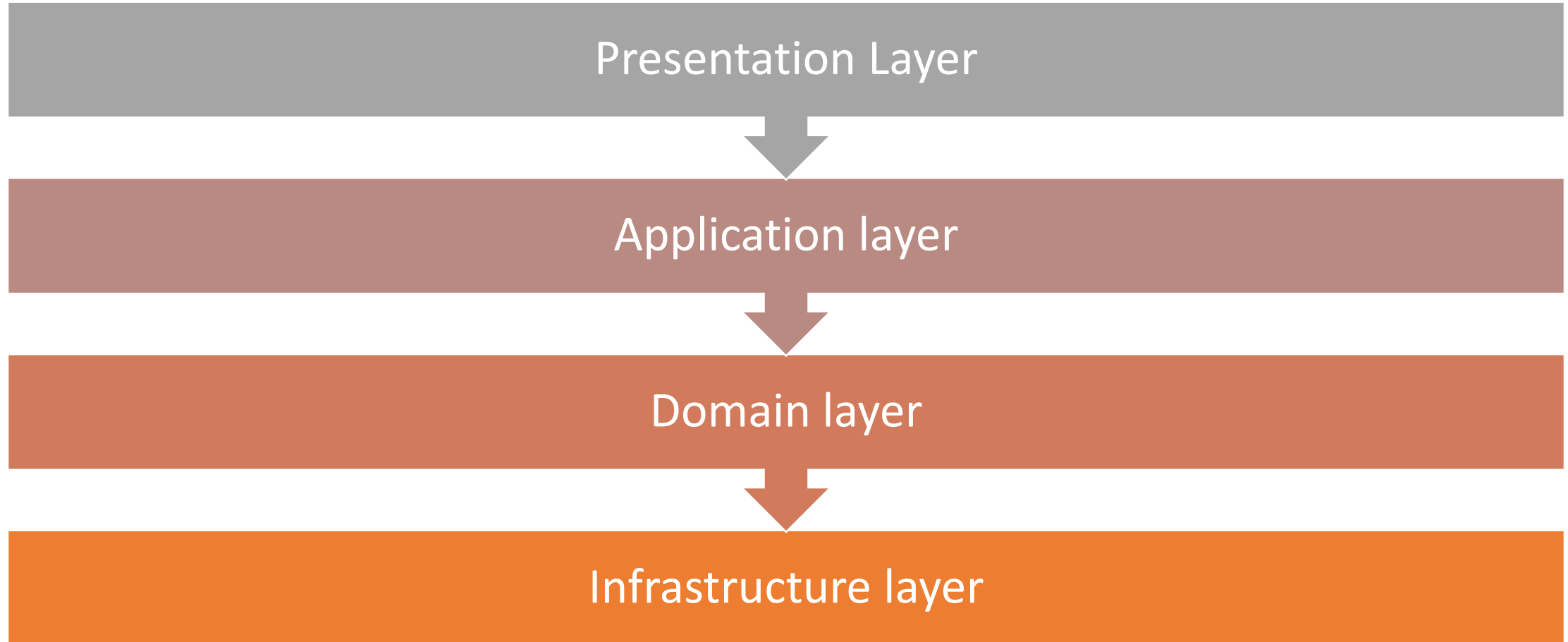


# Архитектура и управление зависимостями

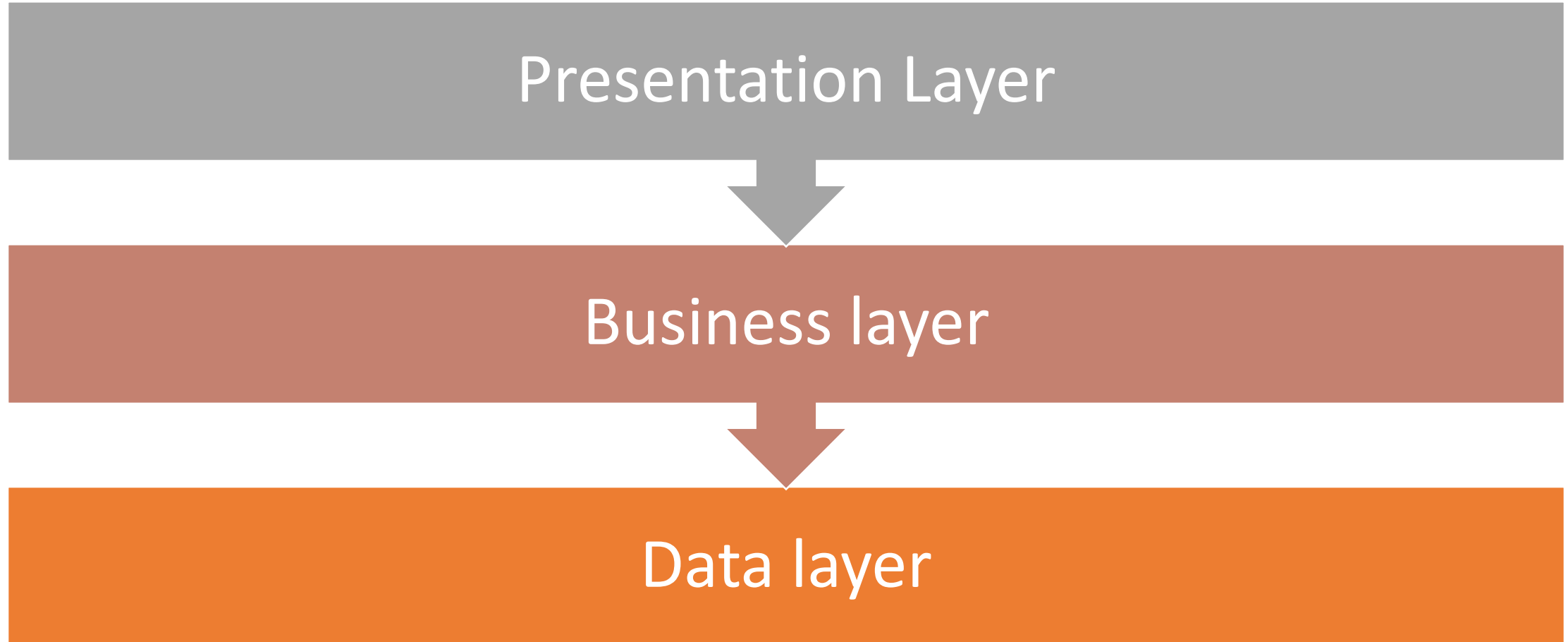
Цель:

**максимально избавить  
доменную логику от  
зависимостей**

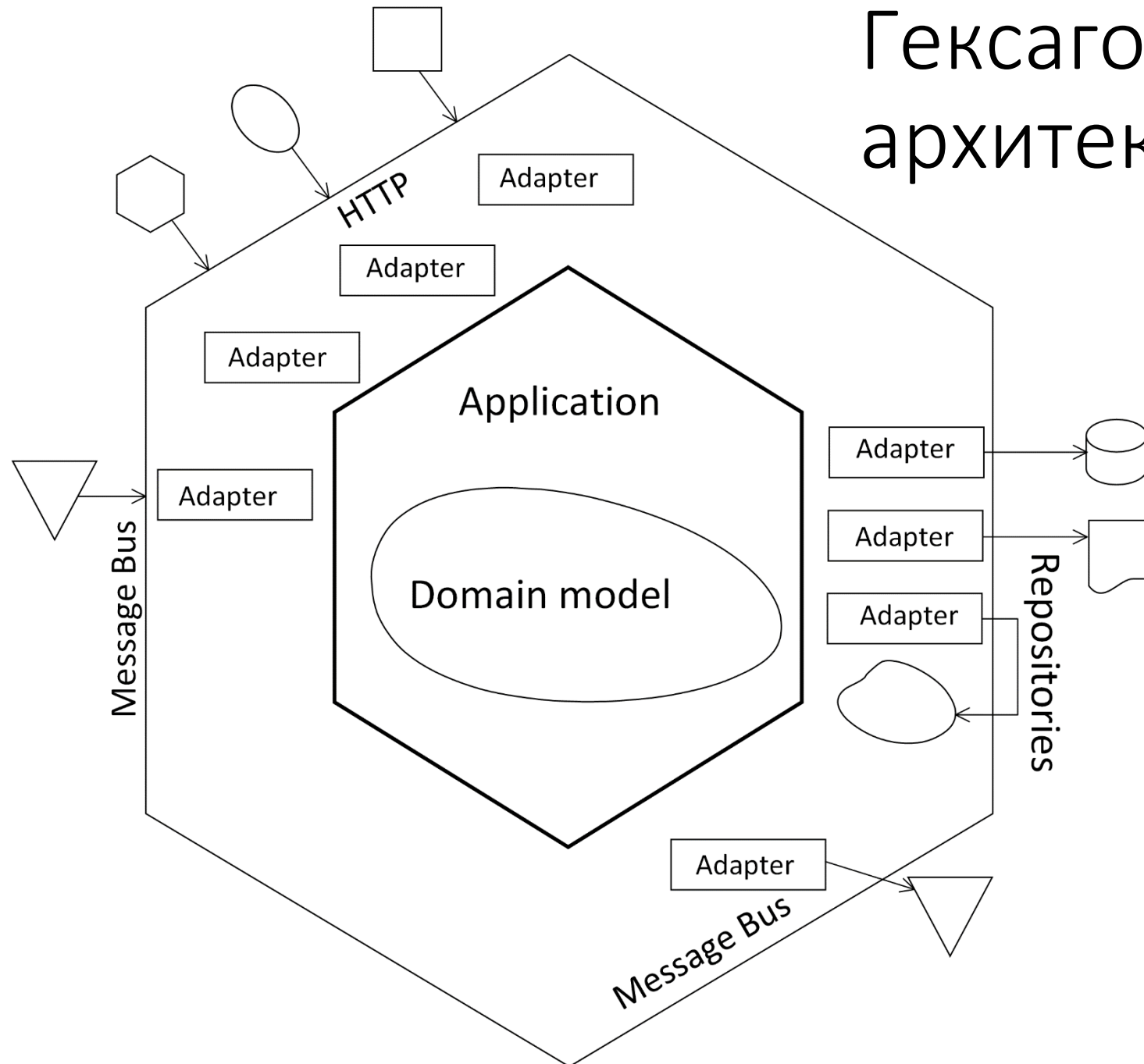
# Архитектура и управление зависимостями



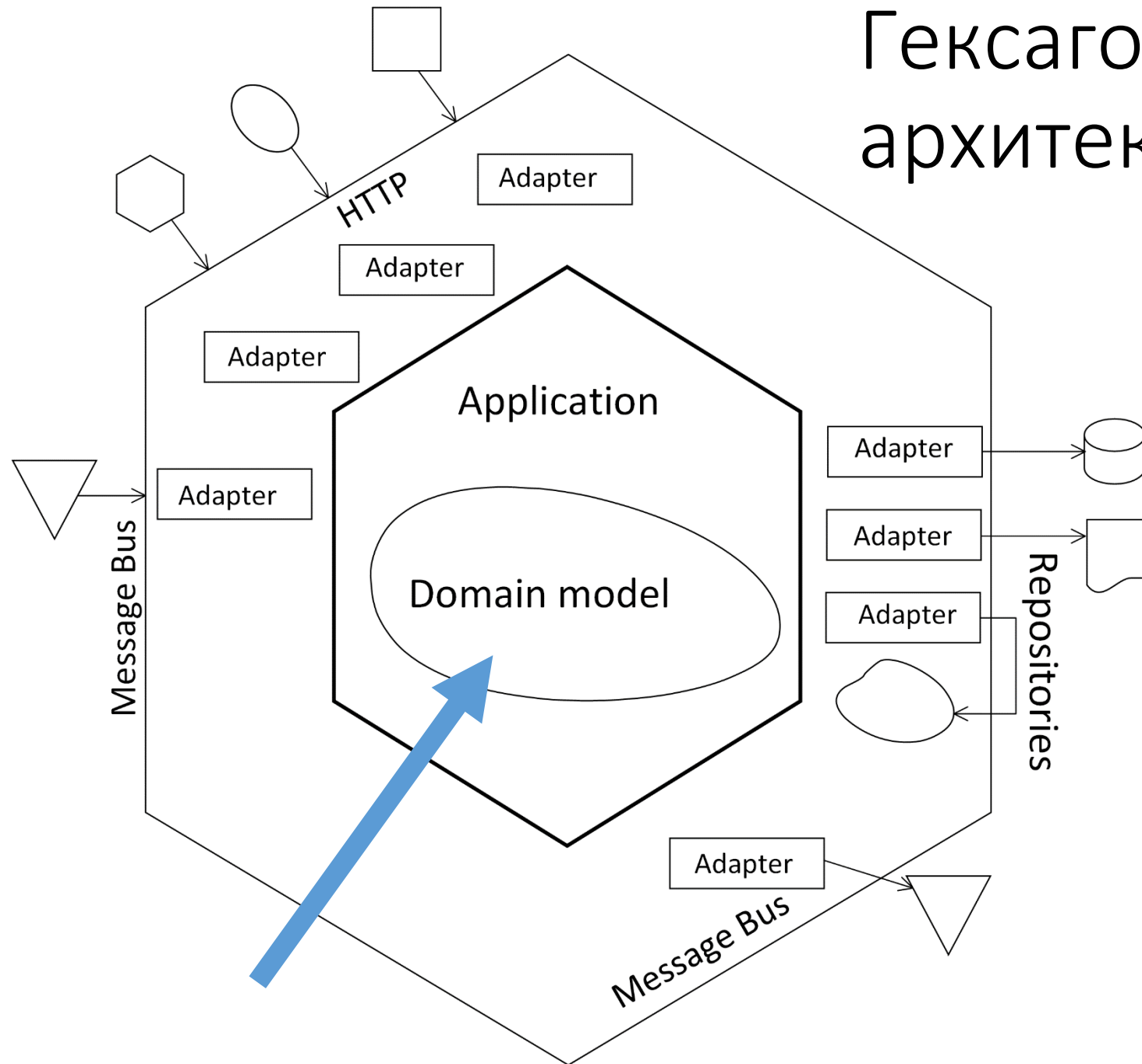
# Архитектура и управление зависимостями



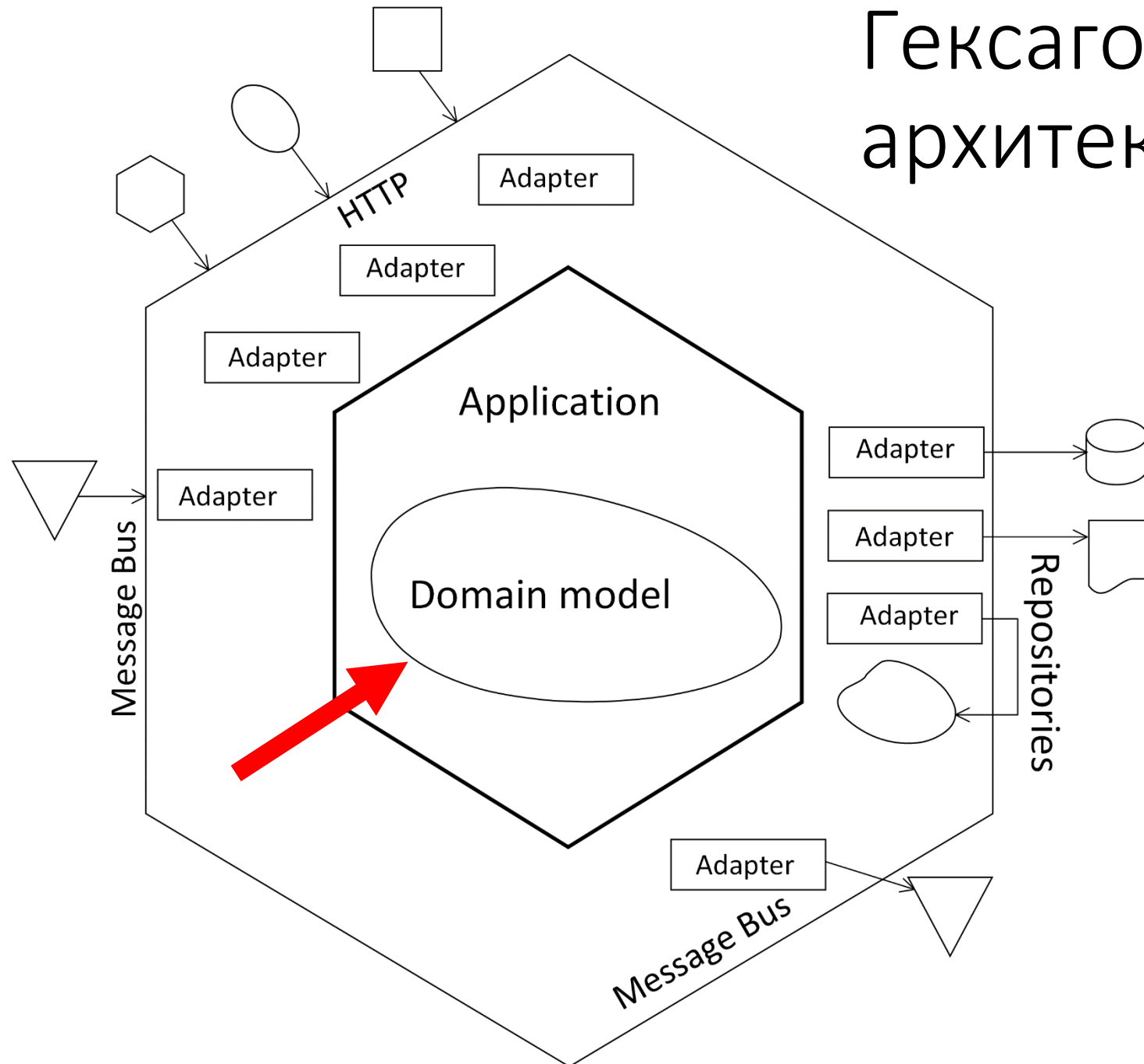
# Гексагональная архитектура



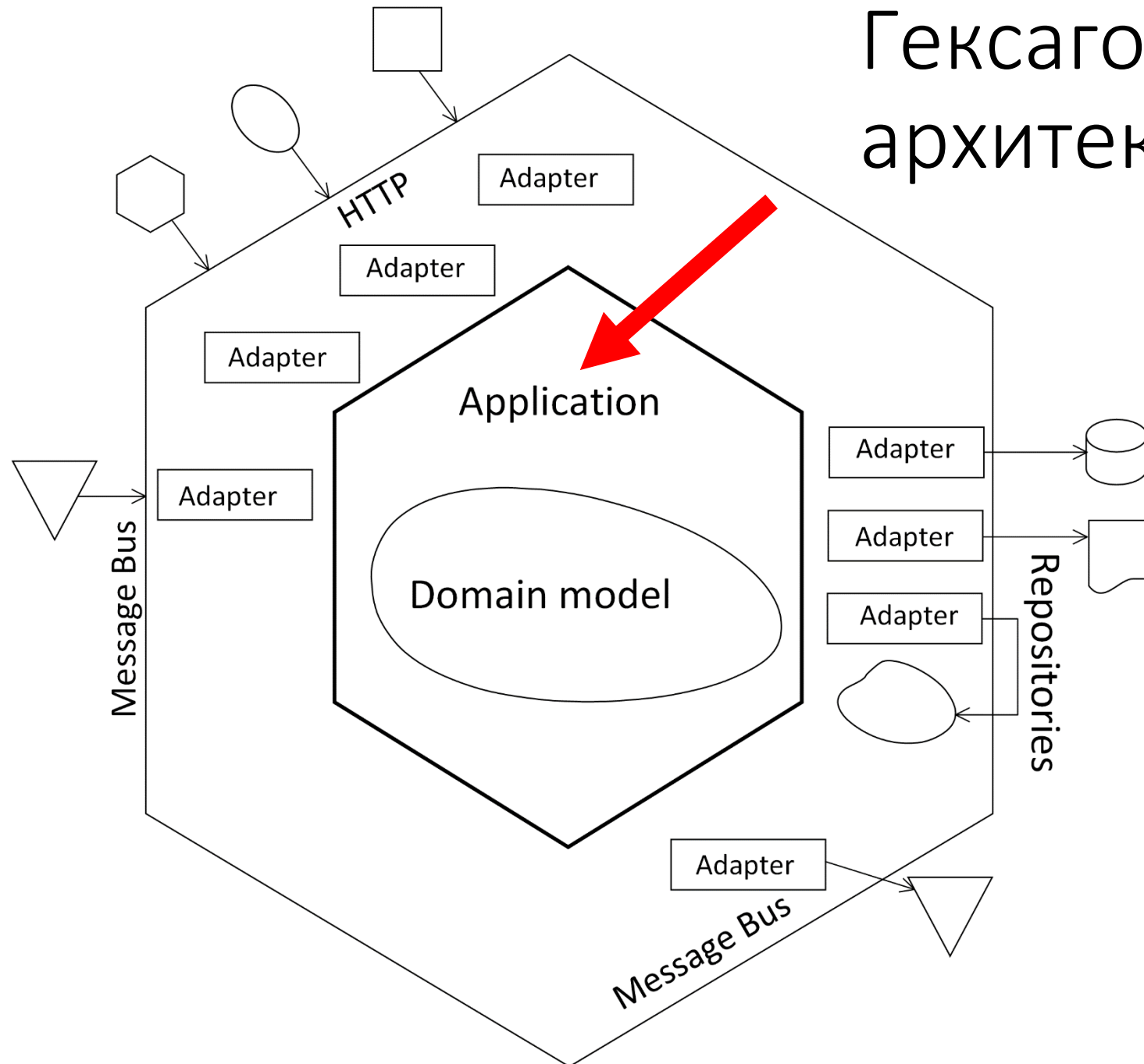
# Гексагональная архитектура



# Гексагональная архитектура

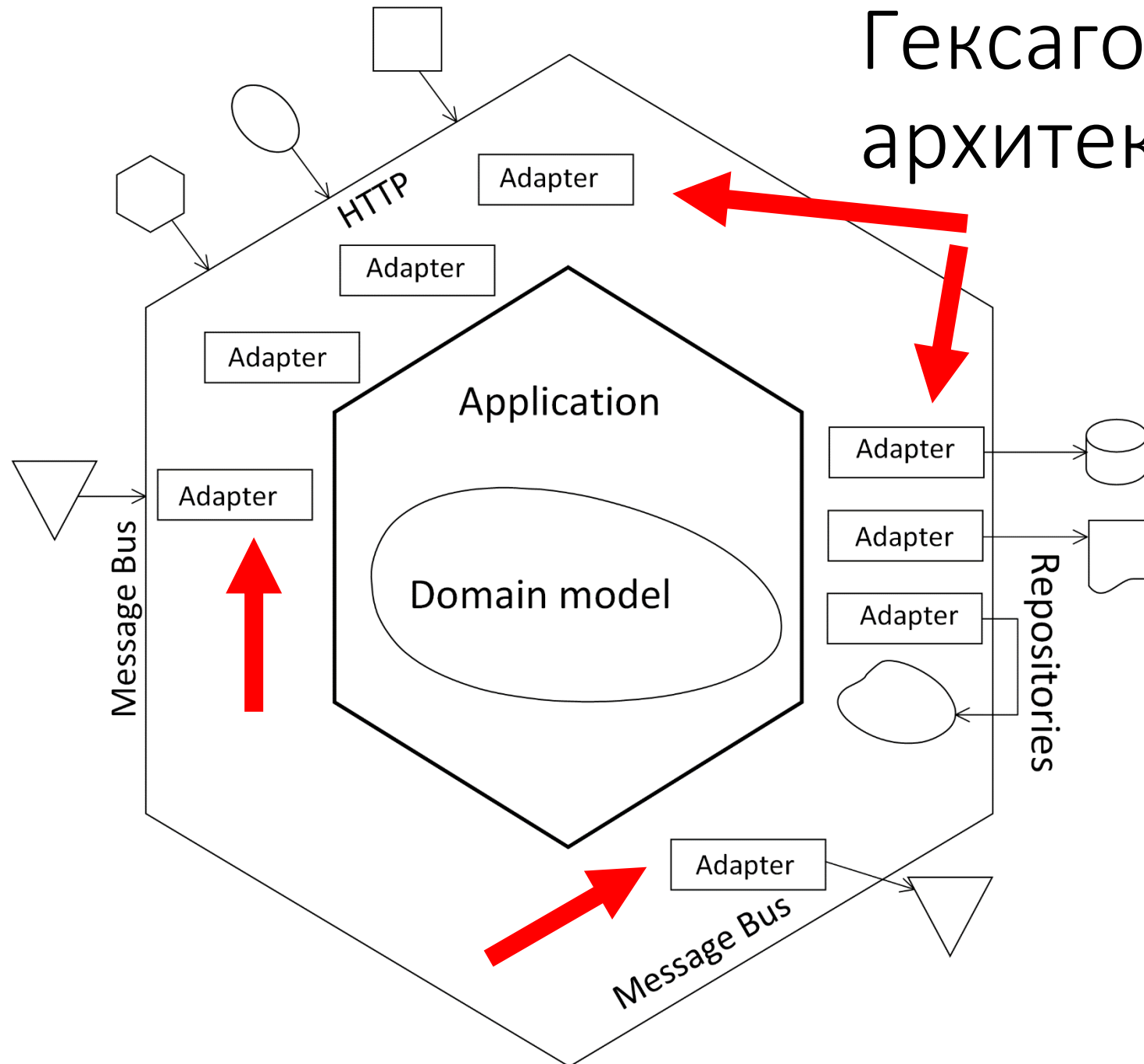


# Гексагональная архитектура

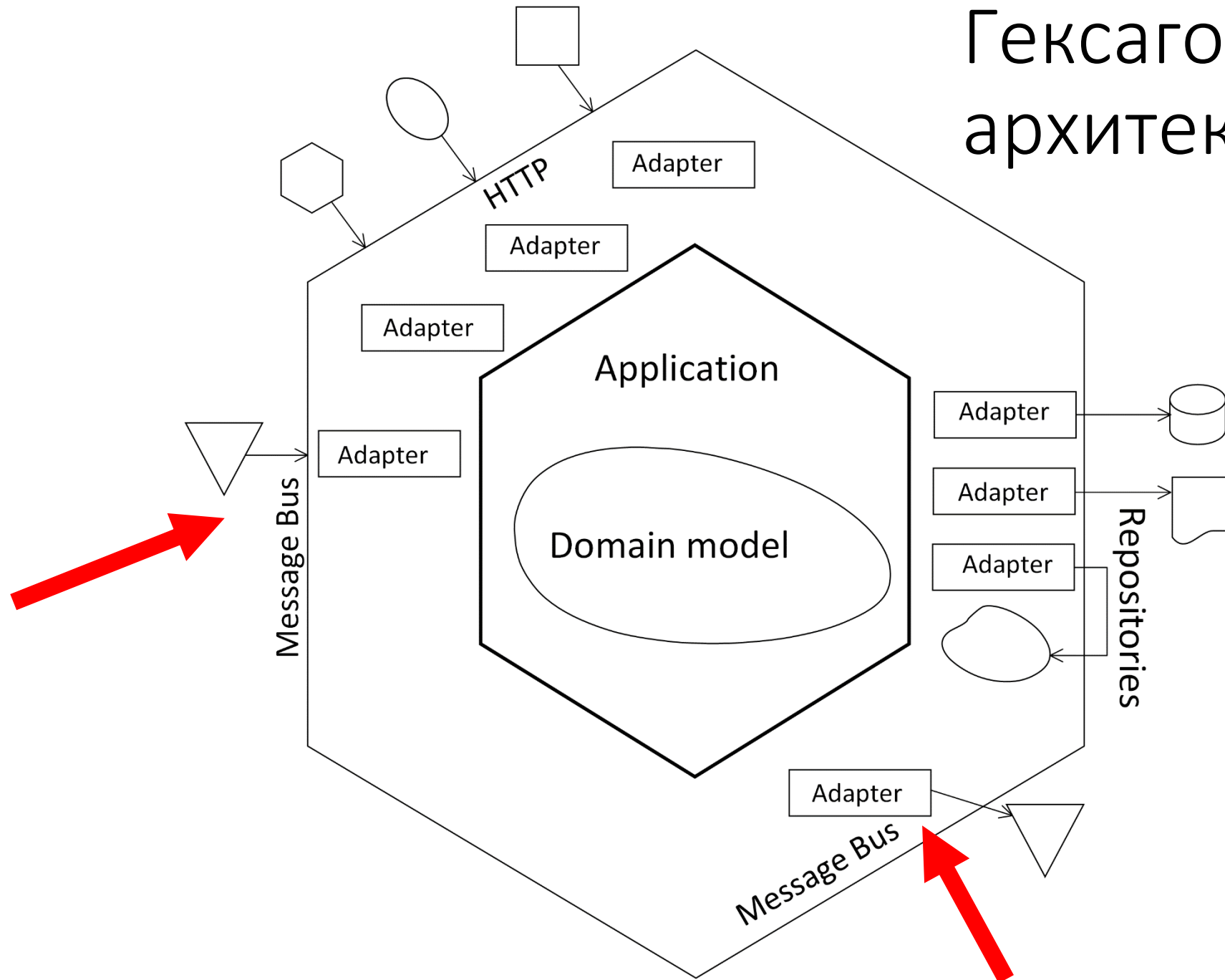




# Гексагональная архитектура



# Гексагональная архитектура



Немного про тактические  
паттерны:

Separated Interface

Еще раз о языке

# Функциональный подход и читабельность кода

# Подытожим

- Общение + общение + общение = ubiquitous language
- Максимально выразительный код бизнес-логики
- Модели ограничиваются контекстами
- Минимум зависимостей для доменной модели

# Полезные ссылки:

## 1. Habrahabr

- <https://habrahabr.ru/users/marshinov> (например [Как мы попробовали DDD, CQRS и Event Sourcing и какие выводы сделали](#))
- <https://habrahabr.ru/users/lair/> - статьи не пишет, но очень много полезных комментариев на тему
- Поиск по тегу DDD
- <https://habrahabr.ru/users/mad/favorites/>

## 2. Блоги

- <https://lostechies.com/jimmybogard/author/jimmybogard/>

## 3. DDD и функциональное программирование

1. <https://fsharpforfunandprofit.com/>

Спасибо за внимание!



# Контакты:

alexey.merson at gmail