

ОНЛАЙН-ОБРАЗОВАНИЕ

# Меня хорошо слышно && видно?

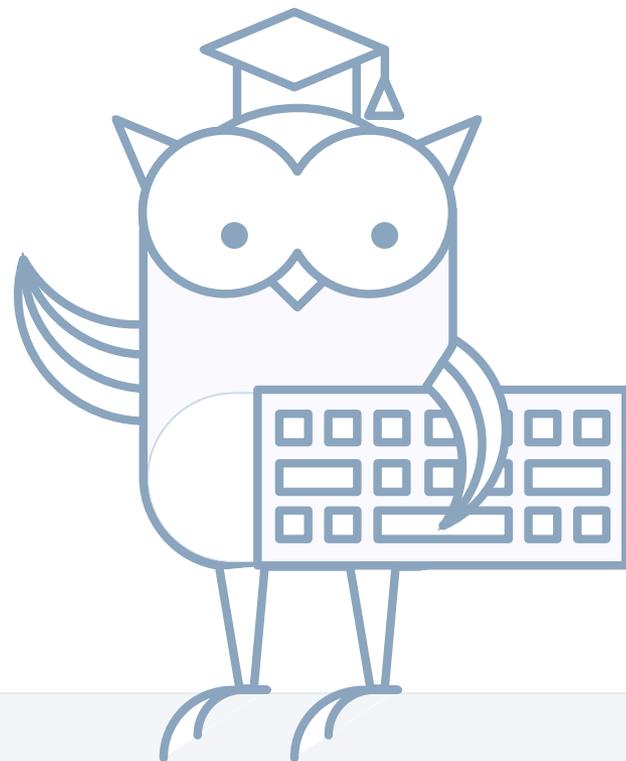


Напишите в чат, если есть проблемы!

Ставьте  если все хорошо



# Шаблоны проектирования



- Обсудить историю возникновения паттернов
- Рассмотреть подробно несколько примеров
- Поговорить про антипаттерны

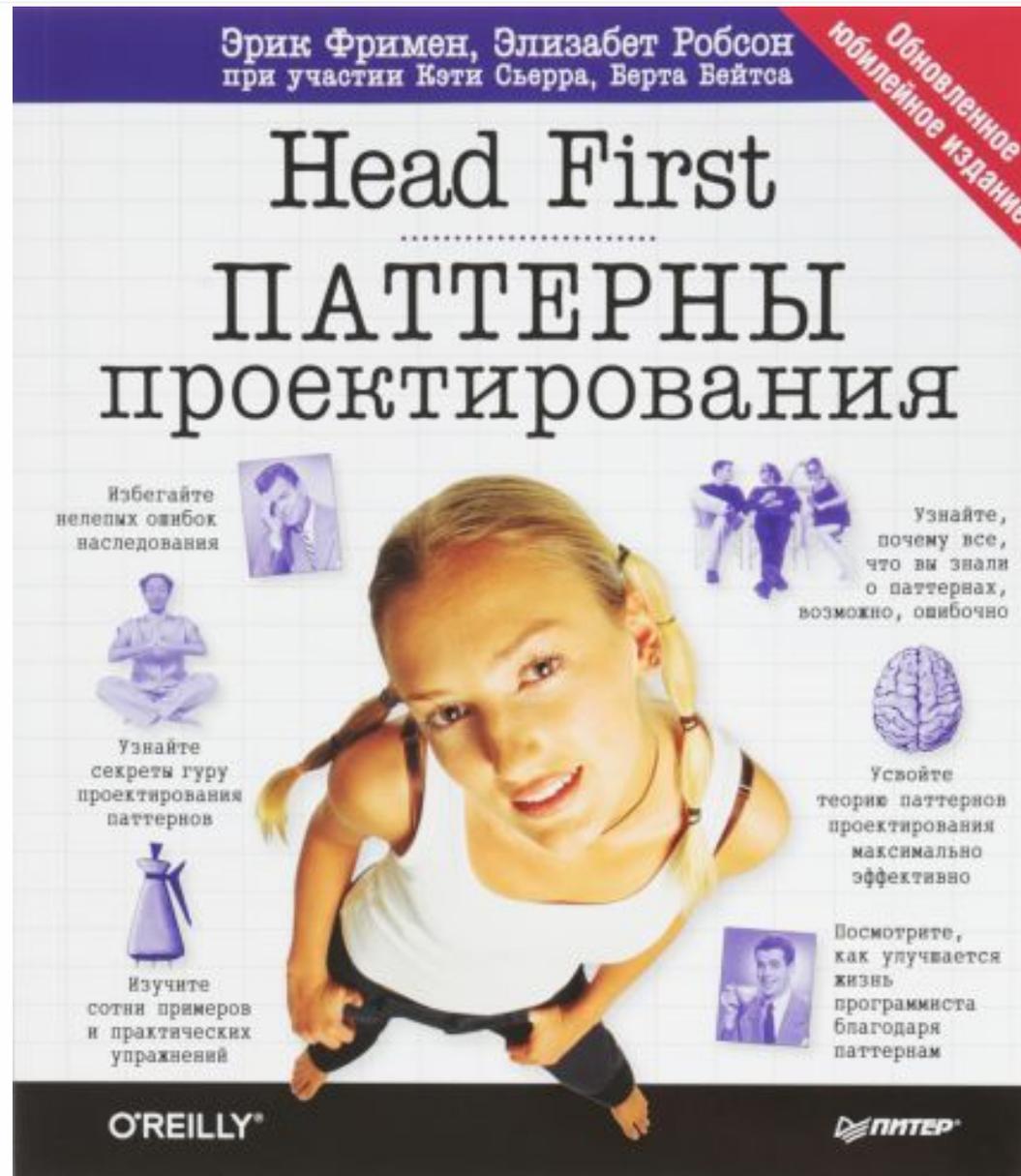
**01**

**История развития  
паттернов**



- Архитектурные
- Дизайн
- Идиомы





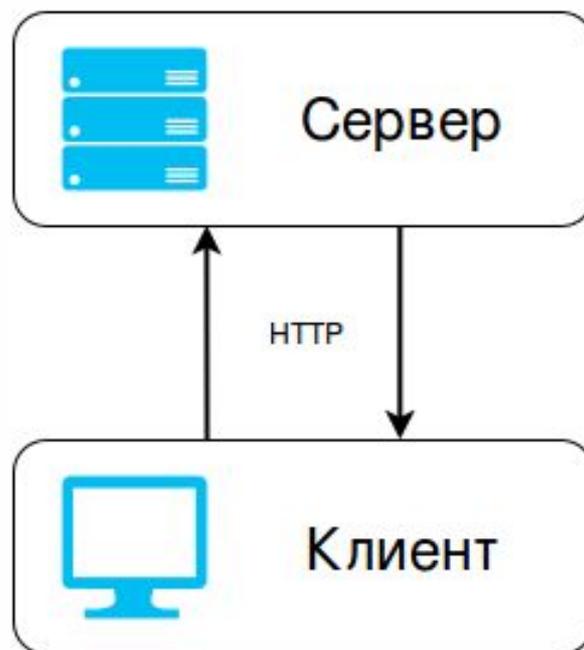
- Проверенные решения
- Стандартизация кода
- Общий словарь (коммуникация)

- Паттерны – не серебряная пуля
- Не стоит пытаться втиснуть паттерны в код
- Ненужное усложнение

02

Паттерны

- Информационный эксперт
- Low coupling and high cohesion
- Устойчивость к изменениям



# Куда поместить логику?





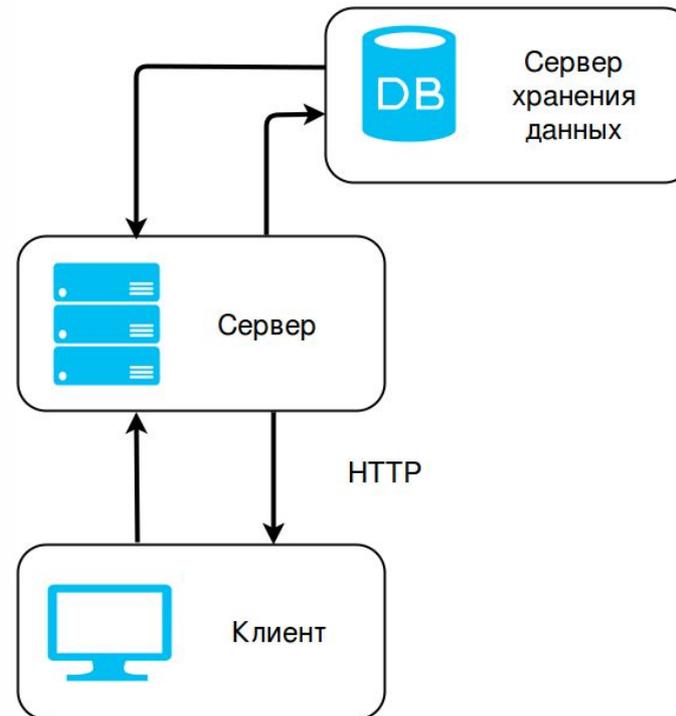
+ не так страшен обрыв связи с сервером

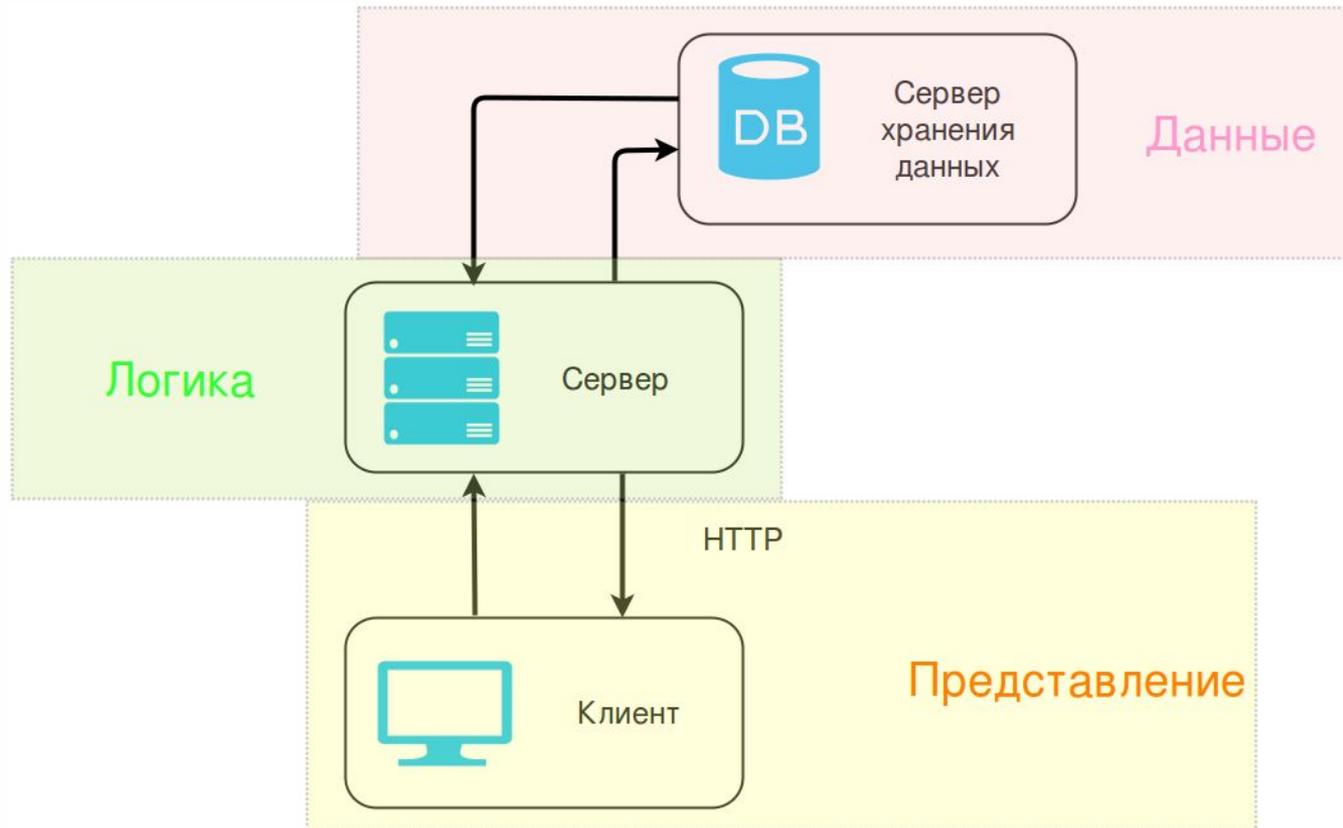
- высокие требования к производительности и проблемы с кроссплатформенностью

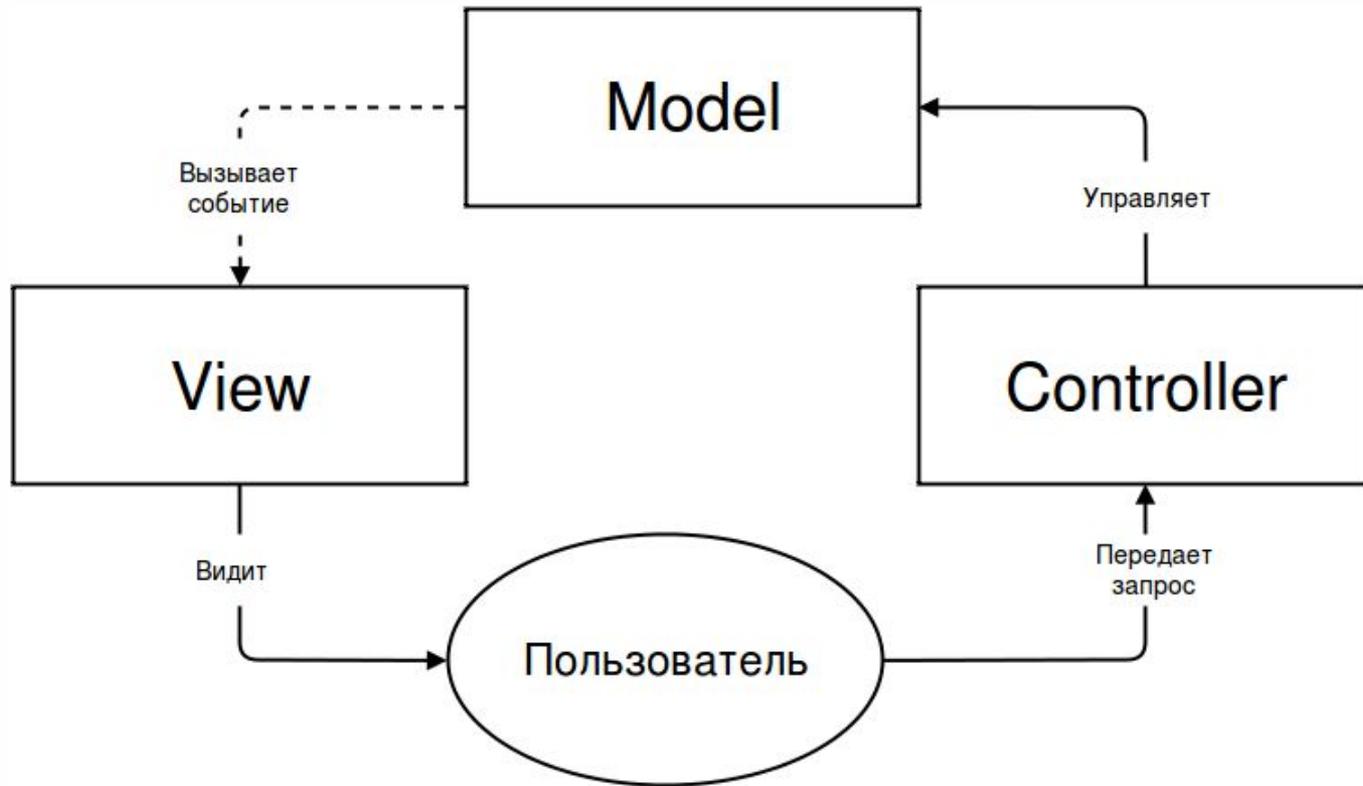


+ клиент простой, легкий и нетребовательный к ресурсам

- высокие требования к серверу и сети из-за нагрузок







## Passive

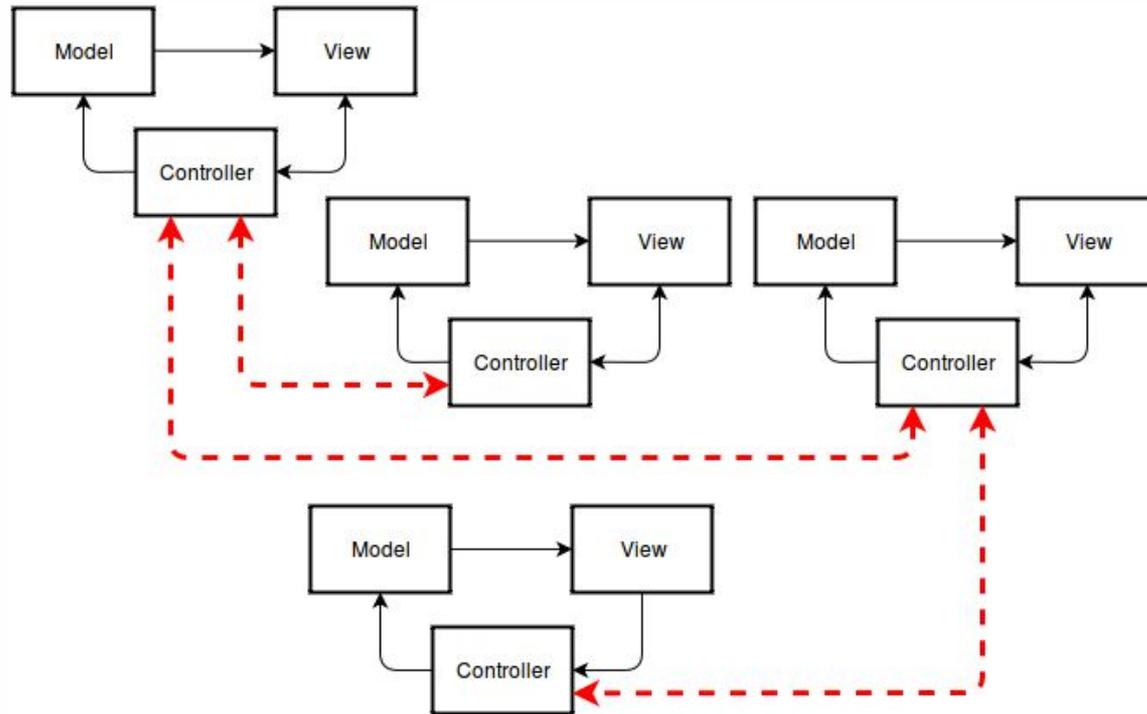
Контроллер становится раздутым и монолитным монстром

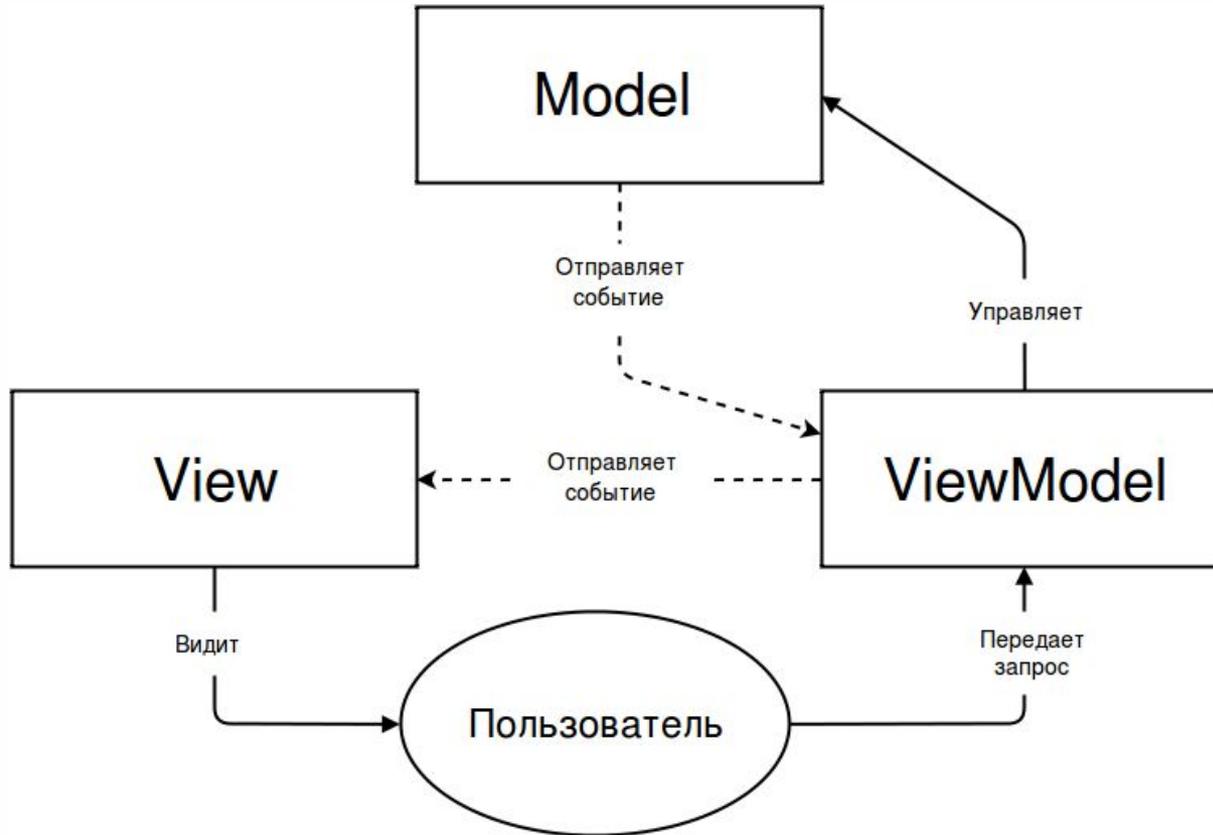
Завязка на фреймворк → ухудшение тестируемости и переиспользования кода

## Active

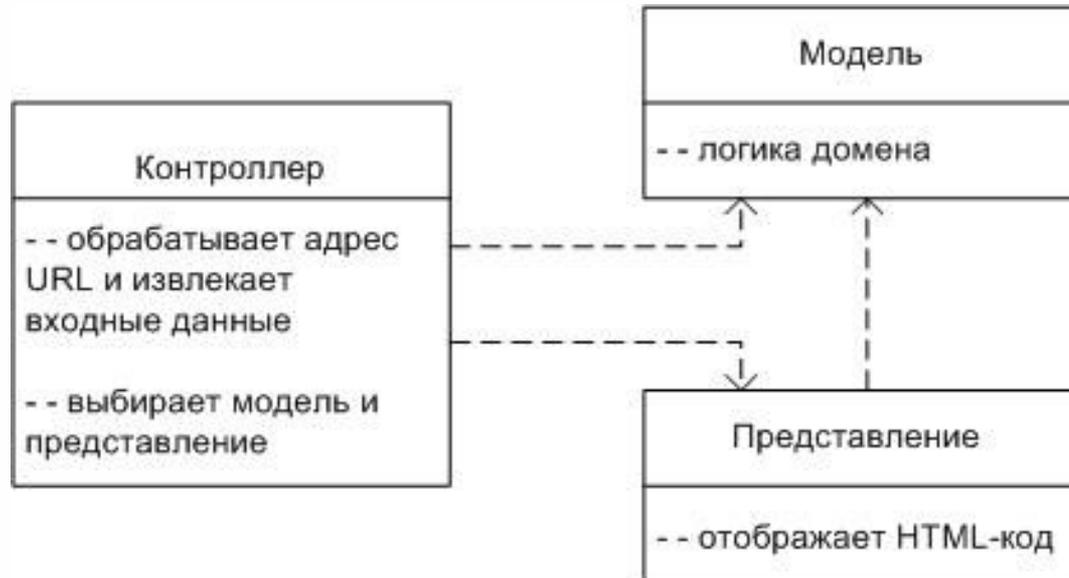
Логика в Модели может быть переиспользуема и лучше тестируется

Контроллер просто связывает Модель с Представлением, не завязывая логику на фреймворк





- Единая точка доступа
- Характерная часть реализации MVC



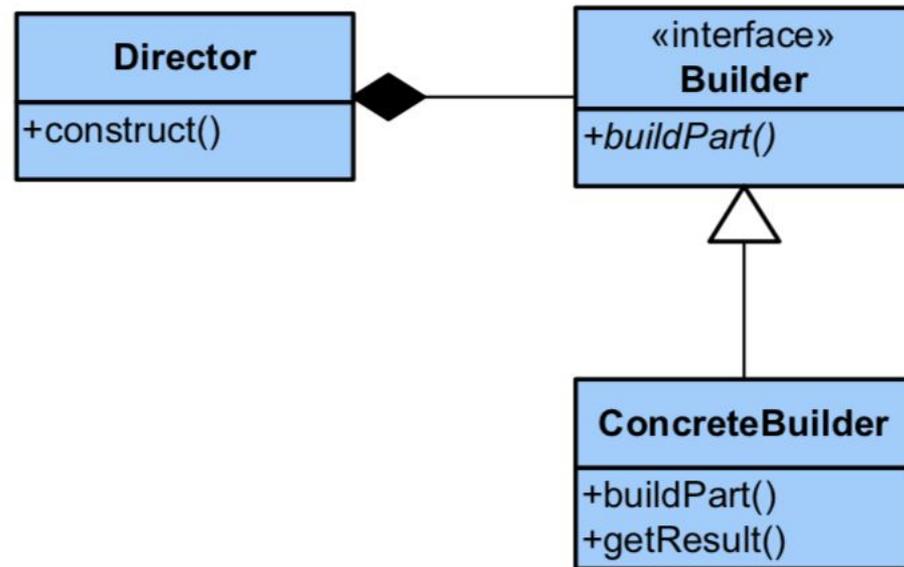
- Порождающие
- Структурные
- Поведенческие

## Builder

**Type:** Creational

**What it is:**

Separate the construction of a complex object from its representing so that the same construction process can create different representations.



## Singleton

**Type:** Creational

**What it is:**

Ensure a class only has one instance and provide a global point of access to it.

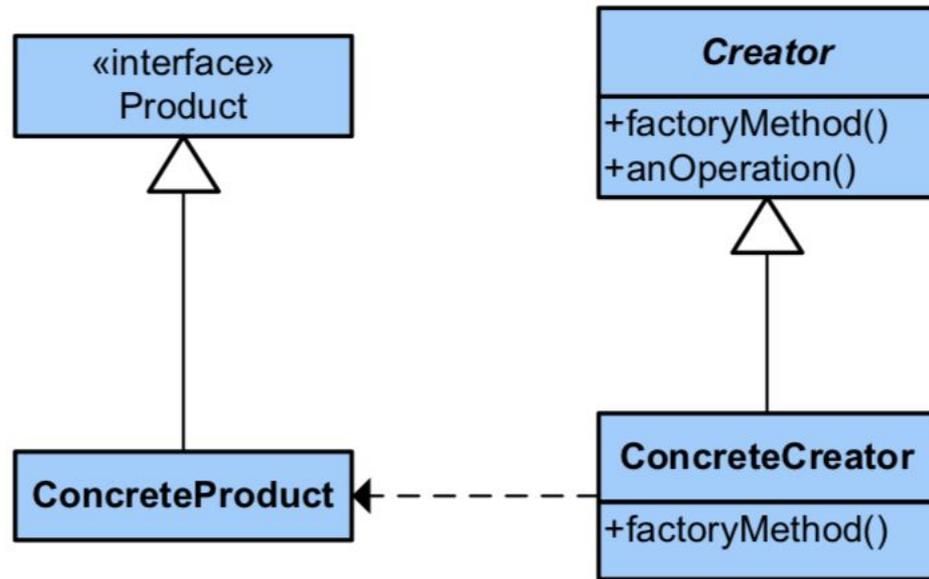
Singleton
-static uniqueInstance -singletonData
+static instance() +SingletonOperation()

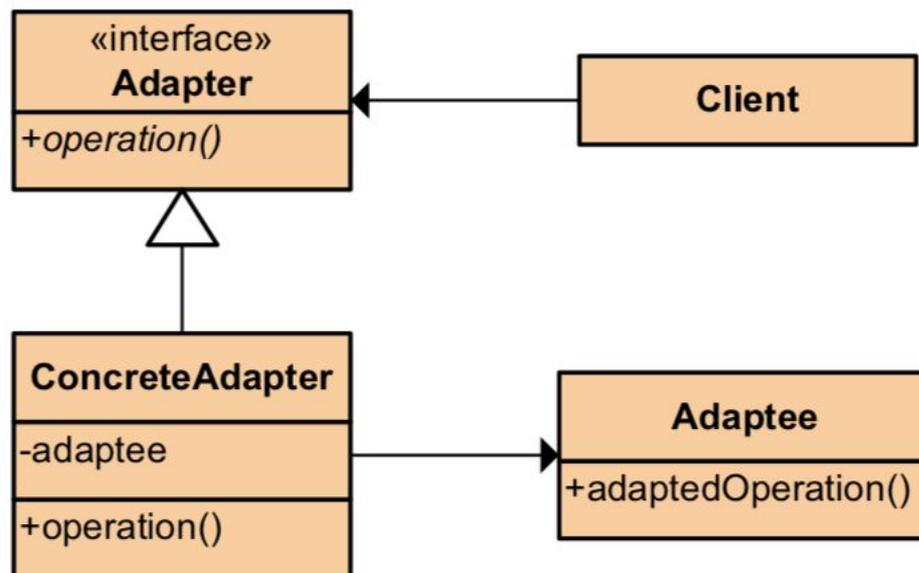
## Factory Method

**Type:** Creational

**What it is:**

Define an interface for creating an object, but let subclasses decide which class to instantiate. Lets a class defer instantiation to subclasses.





## Adapter

Type: Structural

### What it is:

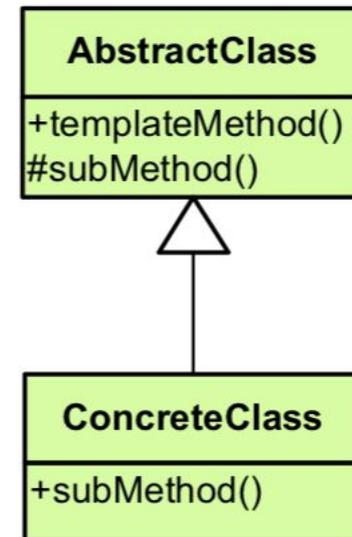
Convert the interface of a class into another interface clients expect. Lets classes work together that couldn't otherwise because of incompatible interfaces.

## Template Method

**Type:** Behavioral

**What it is:**

Define the skeleton of an algorithm in an operation, deferring some steps to subclasses. Lets subclasses redefine certain steps of an algorithm without changing the algorithm's structure.



- Итератор
- Стратегия
- Наблюдатель
  
- ActiveRecord vs Repository

**03**

**Антипаттерны**

Решение:

- Отдельный репозиторий и использование его в качестве внешних зависимостей
- Вынесение кода в библиотеки, утилиты

## Решение

- Статические анализаторы
- Код-ревью
- Рефакторинг

## Решение

- Использовать принципы разработки SRP и другие
- Рефакторинг

- Магические числа
- Hardcoding
- Сохранение части кода на будущее
- Изобретение колеса



# Александр Пряхин

Email: [av.priakhin@gmail.com](mailto:av.priakhin@gmail.com)

Telegram: [@alex\\_kex](https://www.t.me/@alex_kex)