

Software Engineering Conference Russia  
October 2017, St. Petersburg



# Архитектура клиентского приложения

Кирилл Лебедев, Social Quantum

# О себе

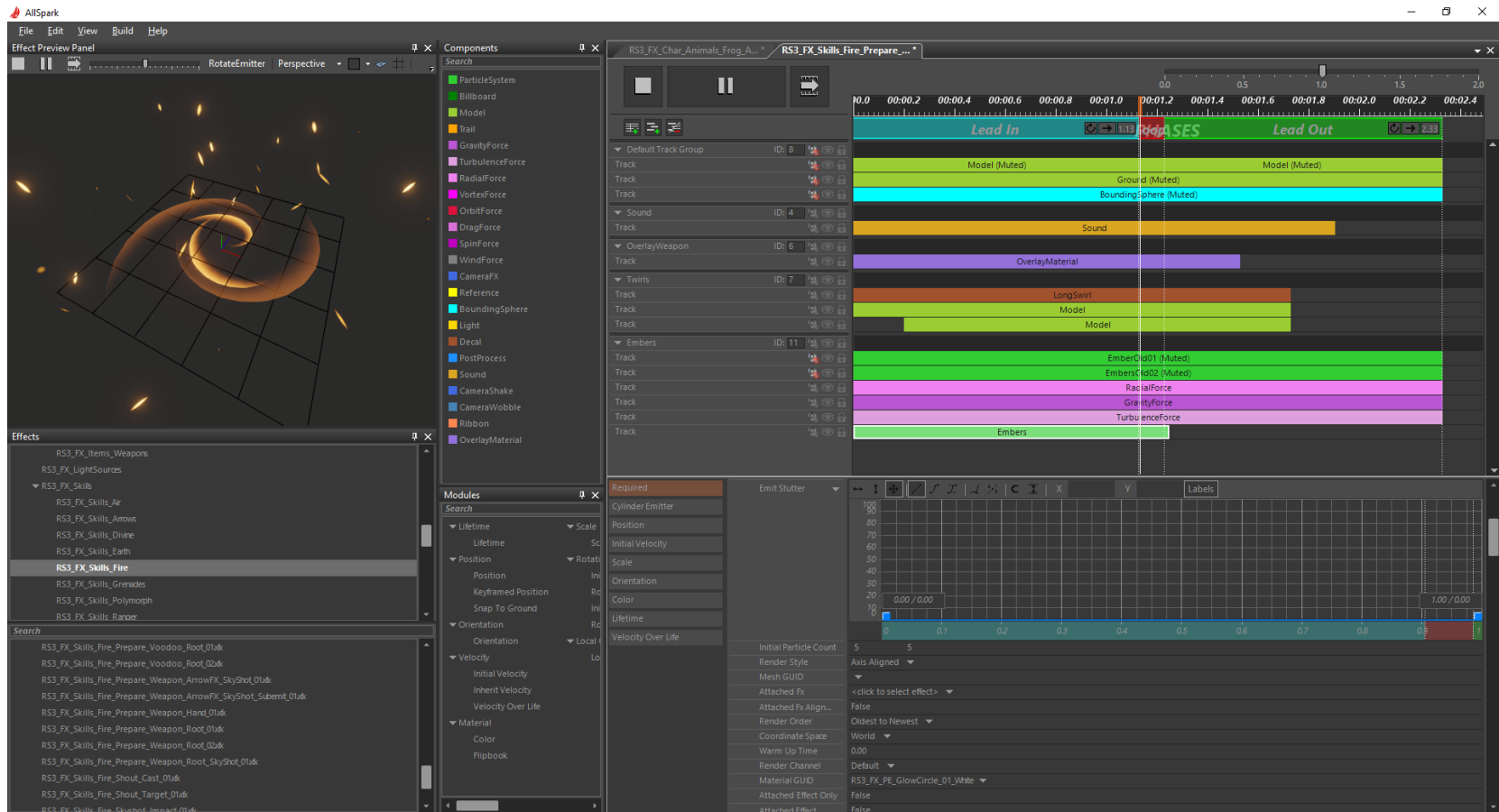


- Руководитель разработки в Social Quantum
- 21 год в ИТ-индустрии
- 11 лет в игровой индустрии
- ~35 успешно завершенных проектов
- ~25 коммерческих продуктов:
  - FIFA Street
  - FIFA 2013
  - NHL 2014
  - FIFA World Cup 2014
  - FIFA 2015
  - Divinity: Original Sin Enhanced Edition
  - Divinity: Original Sin 2
- [habrahabr.ru/users/askofen/](http://habrahabr.ru/users/askofen/)

# Примеры

Программы-инструменты для Larian Studios

# Редактор визуальных эффектов



# Редактор визуальных эффектов (продолжение)

- Предназначен для создания визуальных эффектов
- Напоминает аудиоредактор
- Содержит «ось времени» и «треки»
- На «треках» располагаются «компоненты»
- В качестве «компонентов» выступают:
  - частицы
  - биллборды
  - источники света
  - музыка и звуки
  - силы (тяготения, сопротивления)

# Редактор Genome

The screenshot displays the Genome Editor interface for an animation blueprint. The main workspace shows an animation graph with several nodes and connections. The graph includes nodes for 'Use cached pose', 'Layered Blend per Bone', 'Blend Poses by bool', and 'Blend Poses'. The 'Blend Poses by bool' node has three inputs: 'Active Value', 'True Pose', and 'False Pose'. The 'Blend Poses' node has three outputs: 'Active Value', 'True Pose', and 'False Pose'. The 'Layered Blend per Bone' node has three inputs: 'Base Pose', 'Blend Pose 0', and 'Blend Weights 0'. The 'Use cached pose' nodes are connected to the 'Blend Poses by bool' node. The 'Blend Poses by bool' node is connected to the 'Blend Poses' node. The 'Blend Poses' node is connected to the 'doSneak' node. The 'doSneak' node is connected to the 'Blend Poses' node. The 'Blend Poses by bool' node has two 'True Blend Time' and 'False Blend Time' inputs. The 'Blend Poses' node has two 'True Blend' and 'False Blend' outputs. The 'Blend Poses by bool' node has two 'True Blend Time' and 'False Blend Time' inputs. The 'Blend Poses' node has two 'True Blend' and 'False Blend' outputs. The 'Blend Poses by bool' node has two 'True Blend Time' and 'False Blend Time' inputs. The 'Blend Poses' node has two 'True Blend' and 'False Blend' outputs.

**Animation Preview Editor**

Property	Value
characterSpeed	0.00
movingActionBlend	0.00
doMeleeOrRangedAttack	False
doUse	False
doDodge	False
doHitMagic	False
doIdle	False
doDebuff	False

**Library**

Category	Item
Additional	Get Ragdoll Pose
	Play Animation
	Transform (Modify) Bone
Blends	Apply Additive
	Blend
	Blend Poses by bool
Cached Poses	Blend Poses by Int
	Layered Blend per Bone
	New Save cached pose

**Animation Assignment Panel**

Short name	Animation
1Hattack1	Elves_Hero_Male_ABC_Attack_1H_01
1Hdodge	Elves_Hero_Male_ABC_Dodge_1H_01
1Hdodge_backstab	Elves_Hero_Male_ABC_Dodge_Back_1H_01
1Hdraw	Elves_Hero_Male_ABC_Draw_1H_01
1Hhit	Elves_Hero_Male_ABC_Hit_1H_01
1Hhit_backstab	Elves_Hero_Male_ABC_Hit_Back_1H_01
1Hhit_magic	Elves_Hero_Male_ABC_Hit_1H_02
1Hrun	Elves_Hero_Male_ABC_Fast_1H_01
1Hstatus_crippled_01_end	Elves_Hero_Male_ABC_Status_Crippled_1H_01_End
1Hstatus_crippled_01_loop	Elves_Hero_Male_ABC_Status_Crippled_1H_01_Loop
1Hstatus_crippled_01_start	Elves_Hero_Male_ABC_Status_Crippled_1H_01_Start
1Hstill	Elves_Hero_Male_ABC_Still_1H_01
1Hstow	Elves_Hero_Male_ABC_Stow_1H_01

**Blueprint components**

Component	Type
AnimGraph	Blueprint components
Still_Action	Blueprint components
Stand (state)	Blueprint components
Main_Stand_State	Blueprint components
Stand_Idle (state)	Blueprint components
Stand_Still (state)	Blueprint components
Hit_Stand (state)	Blueprint components
Status_Debuff_Start (state)	Blueprint components
Attack (state)	Blueprint components
Spawn (state)	Blueprint components
Talk (state)	Blueprint components
Use_Start (state)	Blueprint components

**Details**

Property	Value
Default Value	0

**Short name list**

Short name	Type
attack_close_01	skill
attack_ground	skill
attack_high_01	skill
attack_low_01	skill
attack1	skill
attack2	skill
attack3	skill
block	skill
die_acid	skill
die_arrow	skill
die_crush	skill
die_dot	skill
die_electrocution	skill
die_frozenshatter	skill
die_incinerate	skill
die_melee	skill
die_pierce	skill
die_slash	skill
dodge	skill
dodge_backstab	skill
draw	skill
hit	skill
hit_backstab	skill
hit_magic	skill
idle1	skill
idle2	skill
idle3	skill

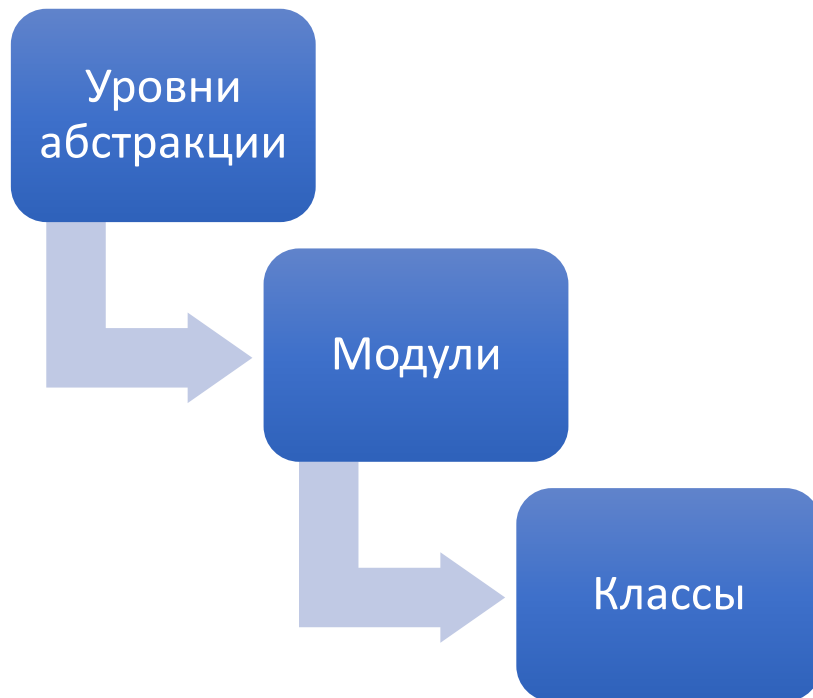
# Редактор Genome (продолжение)

- Предназначен для создания анимационных блюпринтов
- Анимационный блюпринт – это машина состояний персонажа (или набор таких машин)
- Анимационный блюпринт представляет собой иерархический граф
- Редактор позволяет создавать узлы этого графа и соединять их линиями
- Узел – это либо состояние, либо операция (сложение, умножение, проигрывание анимации и т.п.)

Метод

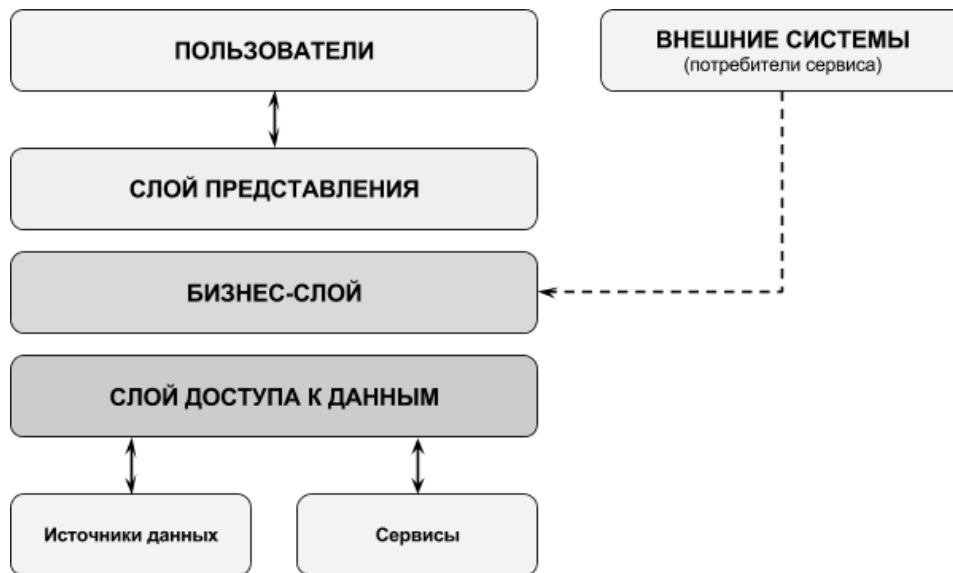


# Структурирование



- Как находить кандидаты в **классы**?
  - *Важный, но **второстепенный** вопрос*
- Начинать структурирование нужно с выявления **уровней абстракции**

# Трехслойная архитектура



- **Три уровня абстракции**
- Данные отделены от представления
- Операции отделены от данных
- Обработка ввода отделена от операций

# К какому слою отнести бизнес-объект?

**К слою доступа к данным?**

- Бизнес-объект считывается из БД

**К слою бизнес-логики?**

- Бизнес-объект участвует в бизнес-операциях

# Пятиуровневая архитектура



- **Пять уровней абстракции**
- Только клиентские приложения
- Не используется СУБД
- Отсутствует сервер приложений
- Основная функция – «редактирование»
- Минимальная бизнес-логика
- Богатый интерфейс пользователя

# Инфраструктура



- Библиотеки для работы с изображениями
- XML- и JSON-парсеры
- Сжатие и шифрование данных
- Контейнеры, деревья и алгоритмы (STL, boost)
- Вспомогательные утилиты

# Инфраструктура (продолжение)

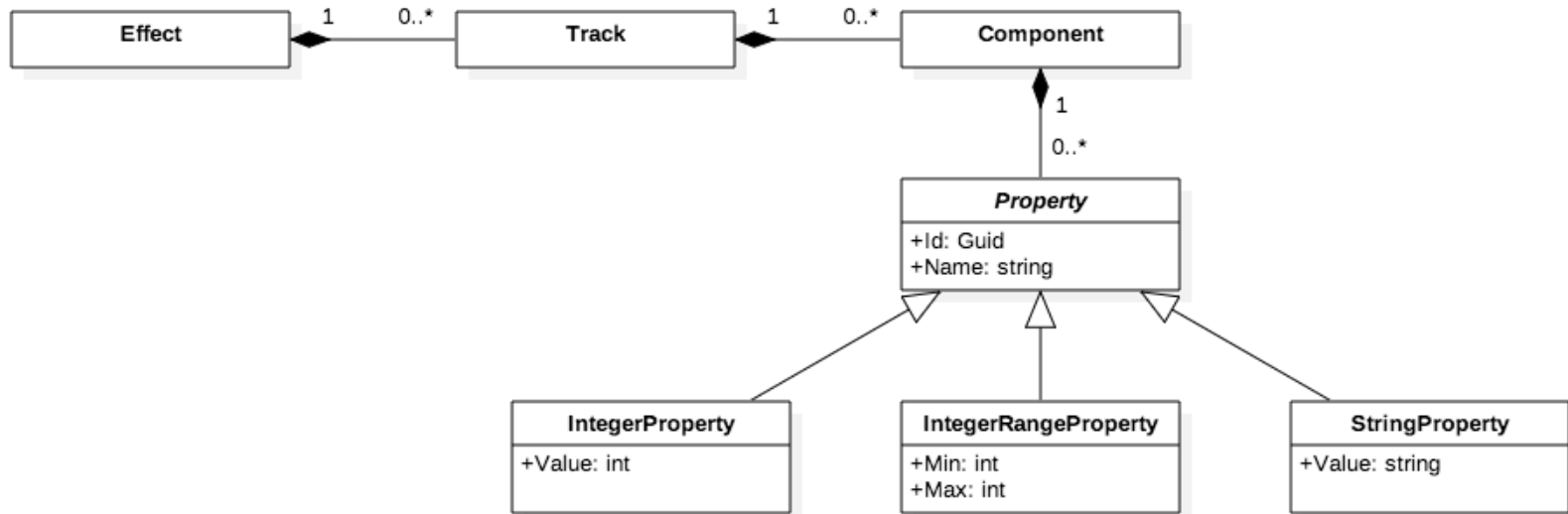
*Все, что **является полезным**, но **трудно упорядочить**,  
нужно помещать в инфраструктурный уровень.*

# Модель данных



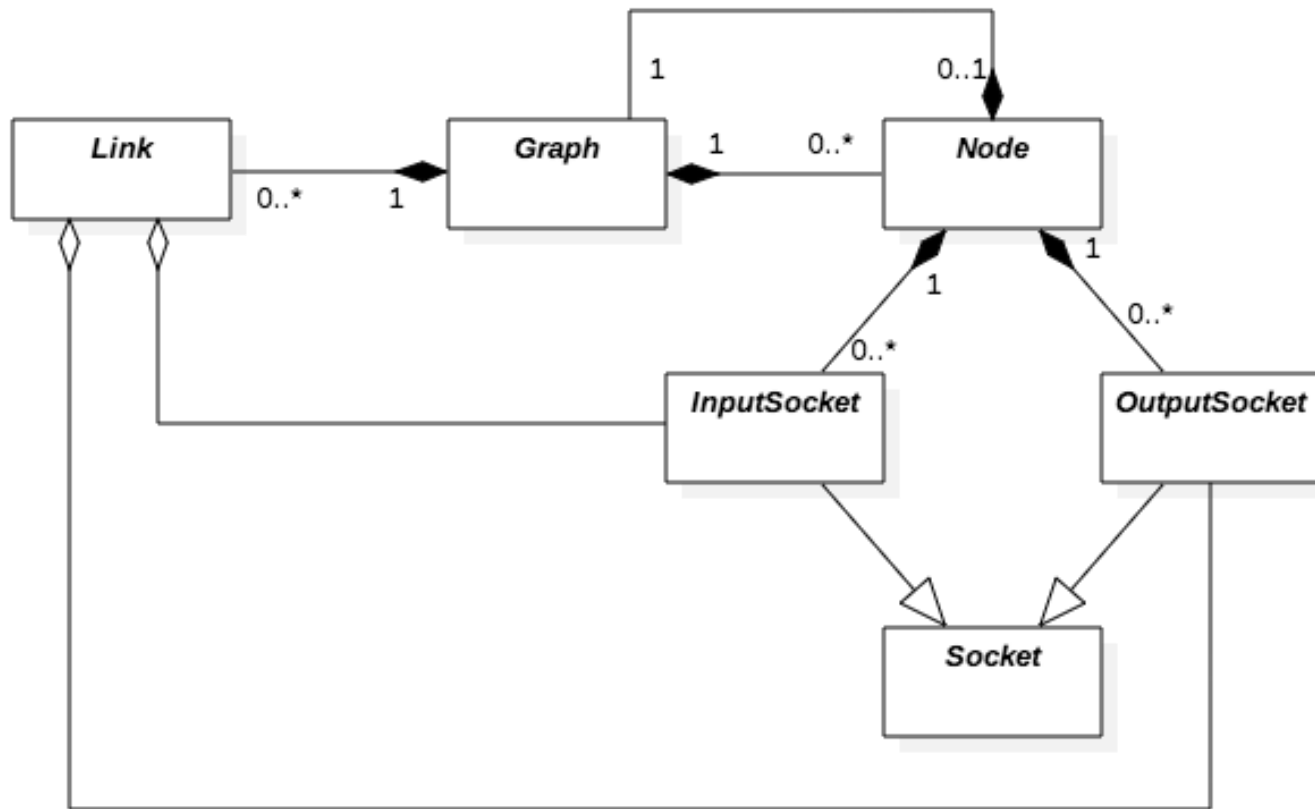
- Объединена с источником данных (JSON- или XML-файл)
- Представляет собой объектную модель предметной области
- Не содержит бизнес-логики
- Сама себя сериализует в XML или JSON

# Модель визуального эффекта

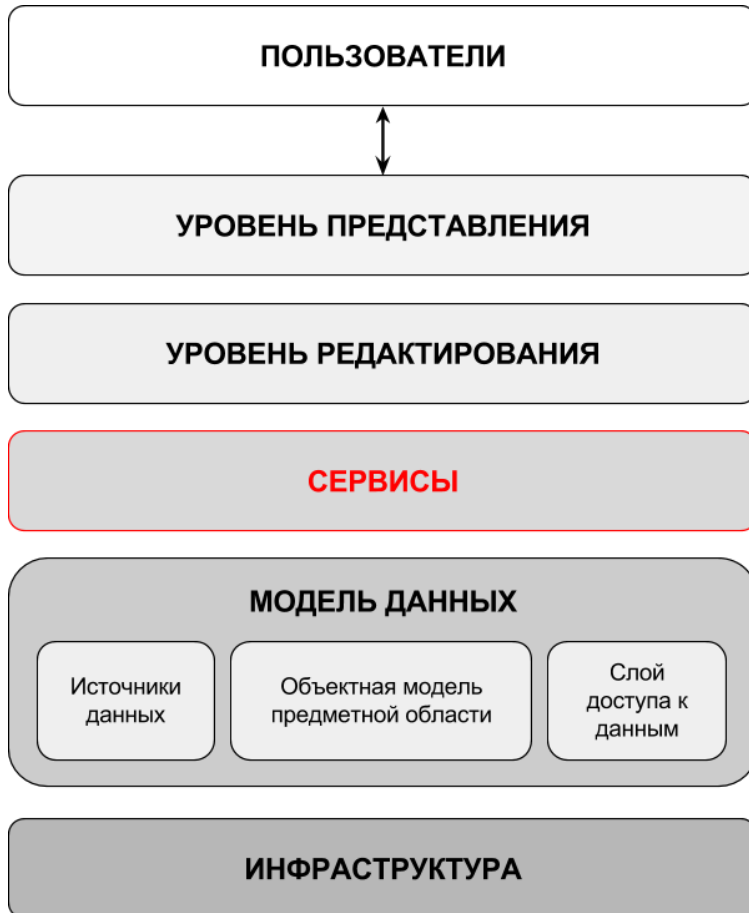




# Модель анимационного блюпринта



# Сервисы



- Набор **функциональных модулей**
- Каждый модуль отвечает за реализацию 1-ой бизнес-функции
- Модули не зависят или слабо зависят друг от друга
- Опирается на модель данных
- Не связан с представлением и обработкой ввода

# Редактор визуальных эффектов



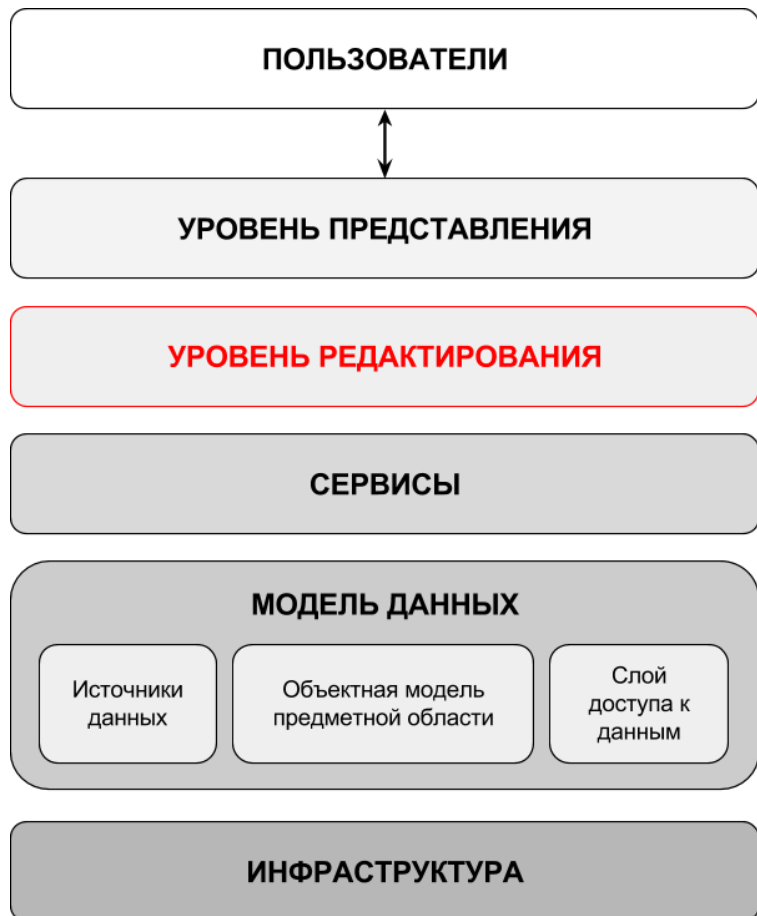
## 1. Компилятор

*Преобразует визуальный эффект в бинарный формат, пригодный для отображения игровым движком*

## 2. Коммуникатор

*Передает скомпилированный визуальный эффект программе для просмотра по протоколу TCP/IP*

# Редактирование

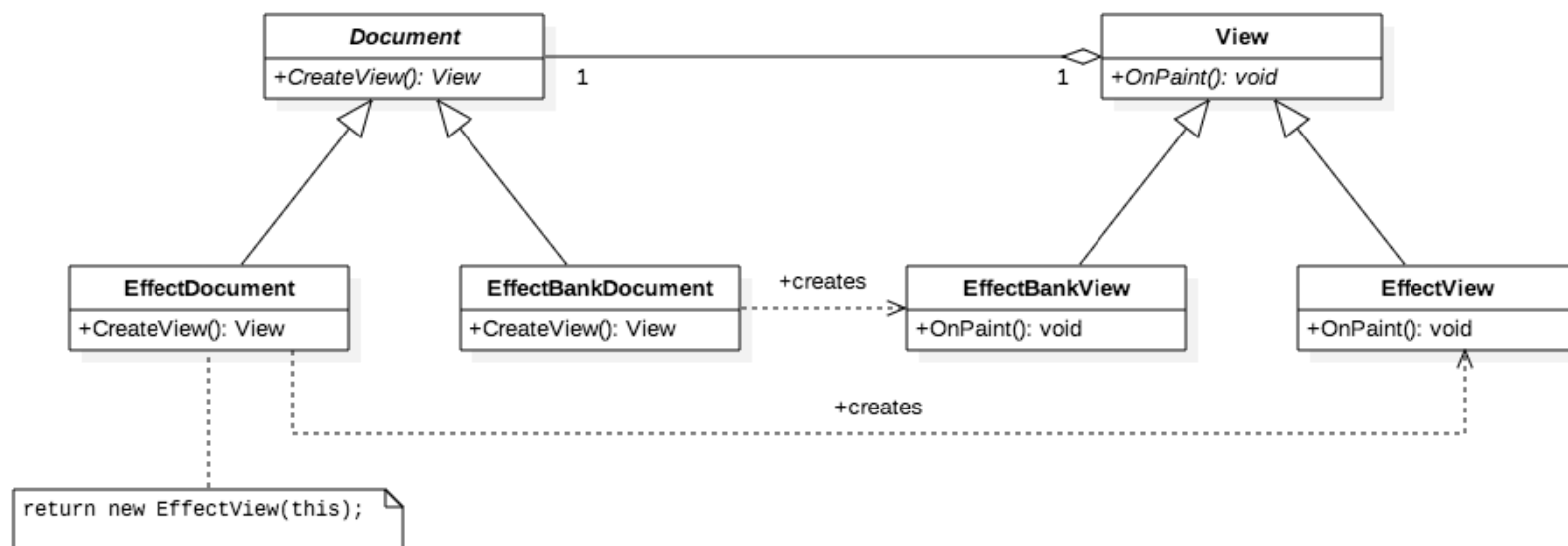


- Характерен для программ-редакторов
- Задает каркас для редактирования модели данных
- Архитектурные концепции редактирования не зависят от визуализации

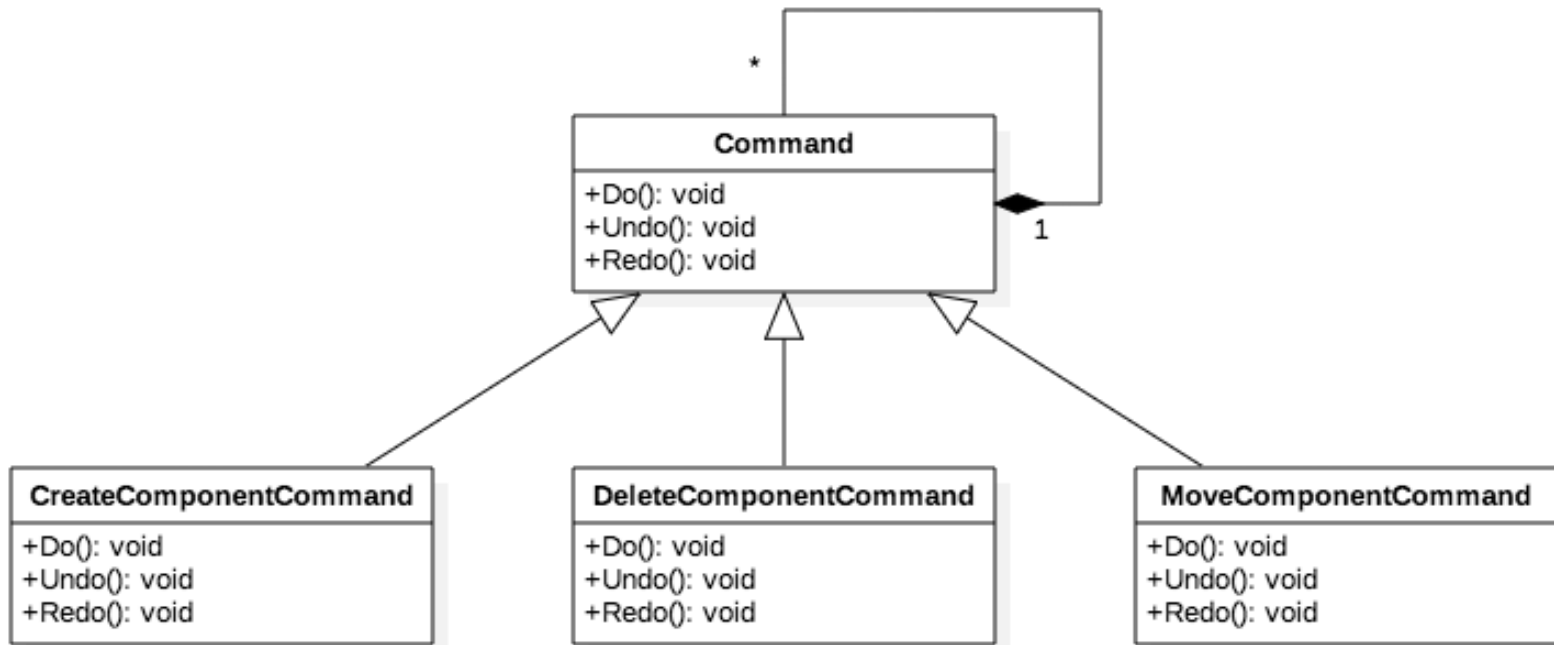
# Базовые паттерны

слоя редактирования

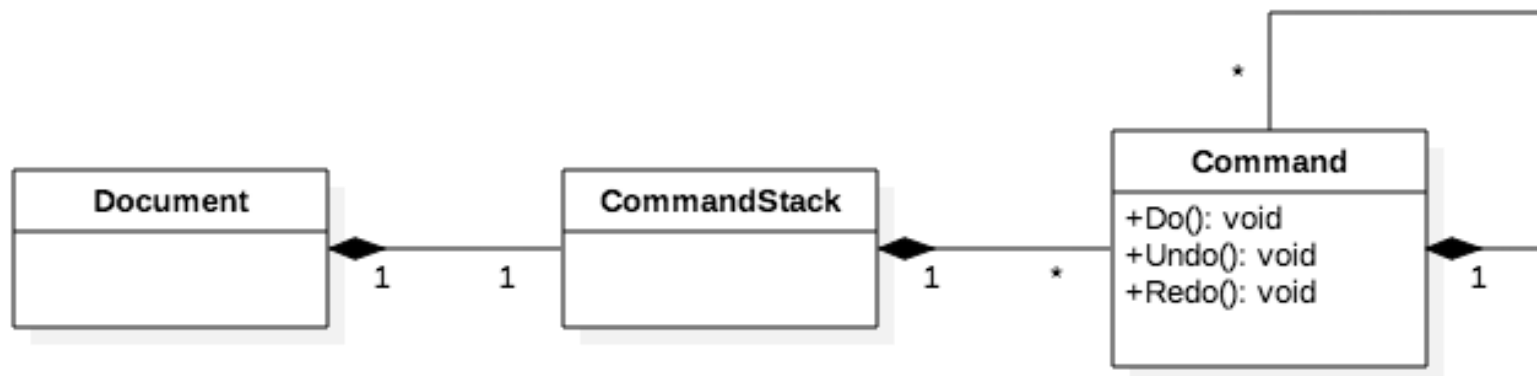
# Архитектура Документ/Вид



# Паттерн «Команда»

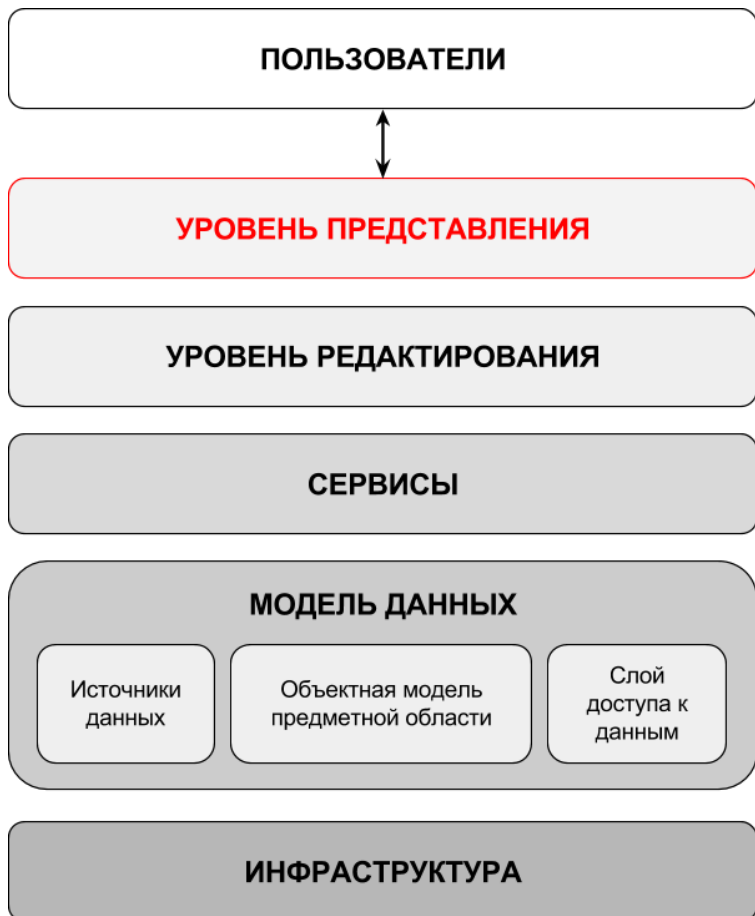


# Undo/Redo стек



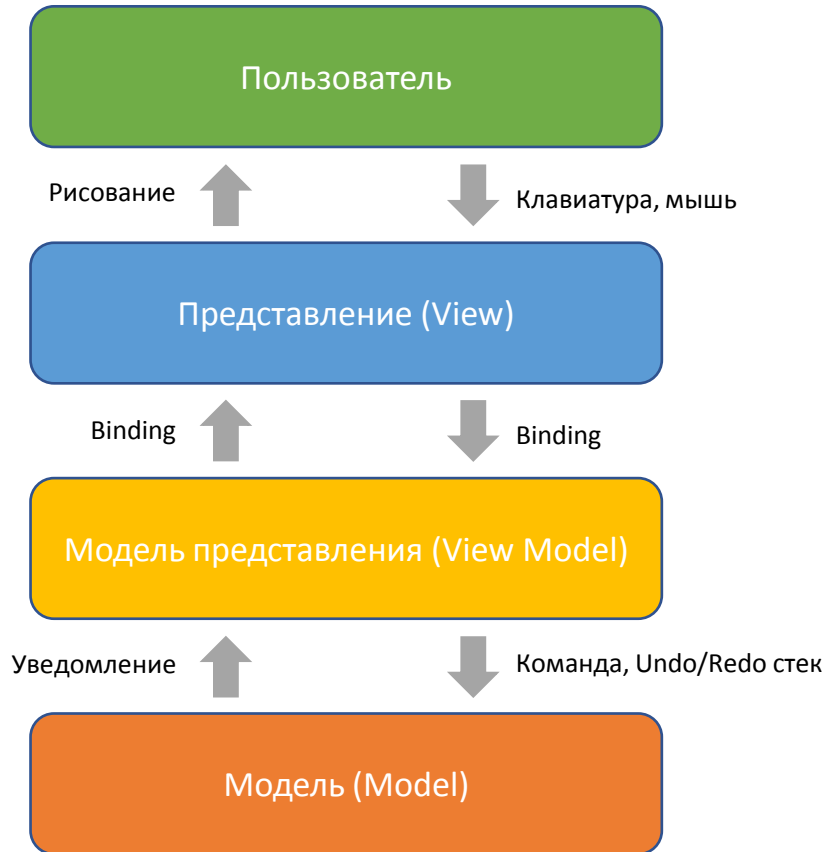


# Представление



- Отвечает за визуализацию данных
- Отвечает за взаимодействие с пользователем
- Изменяет модель данных посредством команд из уровня редактирования

# Паттерн MVVM



- Модель представления синхронизирует данные вида с моделью и наоборот
- Пока пользователь меняет вид – данные расходятся
- Синхронизация происходит после изменений
- Изменения выполняются посредством команд через Undo/Redo стек

# Сравнение архитектур



<b>Инфраструктура</b>	«Свалка» полезных утилит
<b>Модель данных</b>	<ol style="list-style-type: none"><li>1. Источники данных</li><li>2. Слой доступа к данным</li><li>3. Структуры данных (бизнес-объекты)</li></ol>
<b>Сервисы</b>	<ol style="list-style-type: none"><li>1. Операции над моделью данных</li><li>2. Независимые функциональные модули</li></ol>
<b>Уровень редактирования</b>	<ol style="list-style-type: none"><li>1. Архитектура Документ/Вид</li><li>2. Паттерн «Команда»</li><li>3. Undo/Redo стек</li></ol>
<b>Уровень представления</b>	<ol style="list-style-type: none"><li>1. Контролы</li><li>2. Паттерн MVVM</li></ol>

# Вопросы

