

ПЛАТФОРМА ДЛЯ МНОГОАГЕНТНЫХ ИГРОВЫХ АВТОМАТНЫХ СИСТЕМ

**Леонид Столяров (9 класс)
Федор Петряйкин (9 класс)
Никита Уваров (7 класс)**

**Научный руководитель: И.Р. Дединский, МФТИ
Консультант: А.С. Татарinov, NVidia**

ЦЕЛЬ РАБОТЫ

Разработка платформы для многоагентных
игровых автоматных систем

ЗАДАЧИ

- Реализация ядра движка для платформы
- Реализация подсистем движка
 - Графической
 - Физической
 - Звуковой
 - Скриптовой
 - Управления ресурсами
 - Управления объектами
 - Сетевой
- Реализация средств поддержки автоматного программирования в движке
- Создание набора инструментов для работы с движком: редакторов ресурсов, скриптов, конфигураций, а также утилит для работы с моделями

ОПРЕДЕЛЕНИЕ ФУНКЦИОНАЛЬНОСТИ

- Для постановки задач необходимо сначала чётко определить функциональность
- Для этого достаточно представить возможный случай её использования (use case)

Разделение функциональных возможностей по приоритету



Должно быть включено: **100%**

80 - 50%

20 - 0%

ТРЕБОВАНИЯ К ДВИЖКУ

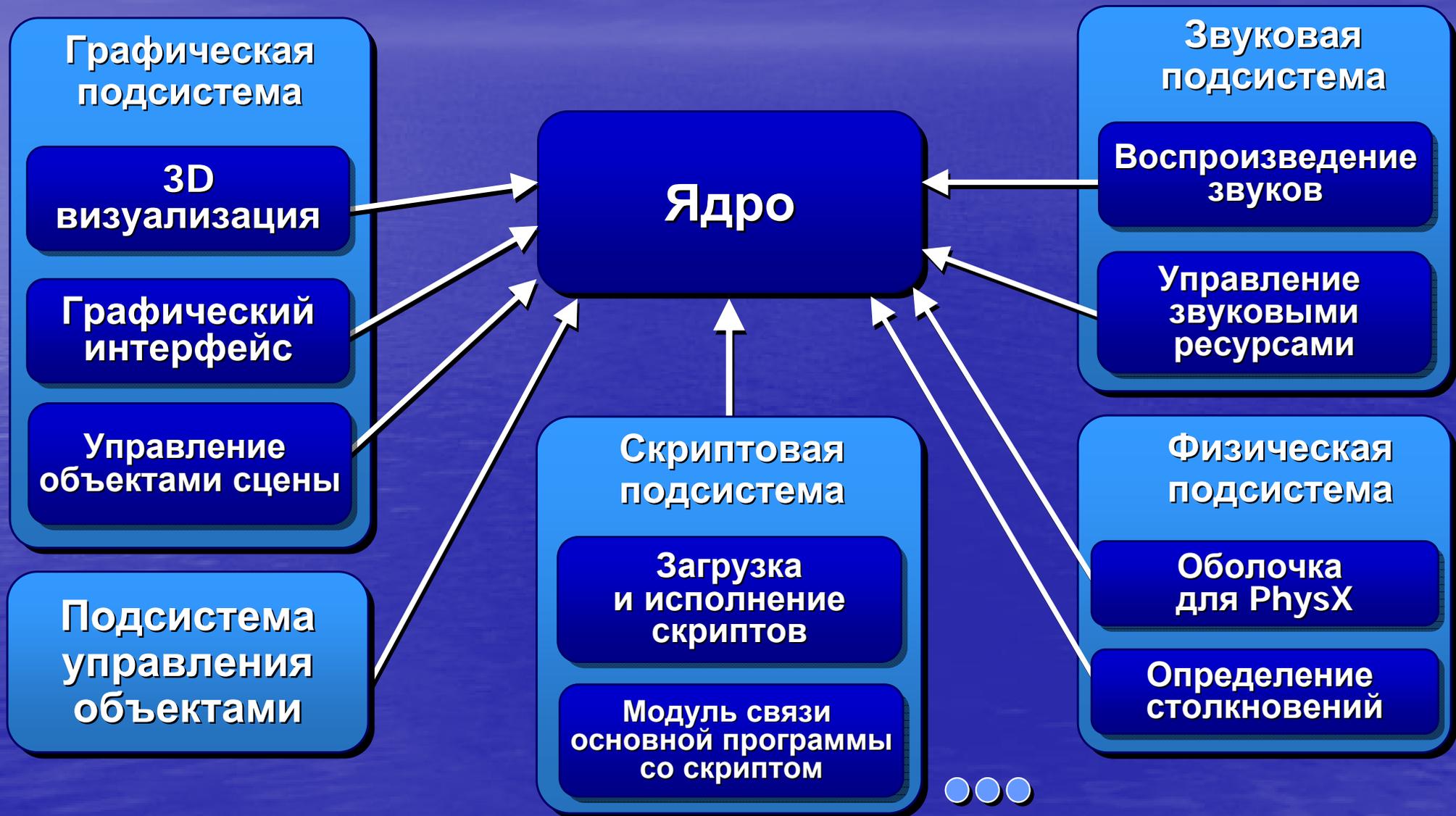
- **Поддержка технологии автоматного программирования**
- **Ориентация на использование в качестве основы автоматной многоагентной системы**
- **Поддержка различных форматов описания конечных автоматов**
- **Высокая производительность**
- **Стабильность работы на различных программных и аппаратных конфигурациях компьютеров**
- **Невысокие системные требования**
- **Поддержка популярных форматов ресурсов**

СРАВНЕНИЕ НЕКОТОРЫХ СВОБОДНЫХ ИГРОВЫХ ДВИЖКОВ

	NeoAxis	OGRE	Unreal Engine*	Existence	Unity 3D
Поддержка АП	нет	нет	частично	нет	нет
Ориентация на MAC	нет	нет	нет	нет	нет
Производительность	низкая	средняя	высокая	высокая	средняя
Стабильность работы	низкая	средняя	высокая	средняя	высокая
Системные требования	средние	средние	высокие	низкие	низкие

АРХИТЕКТУРА ДВИЖКА

Ядро движка синхронизирует работу всех модулей

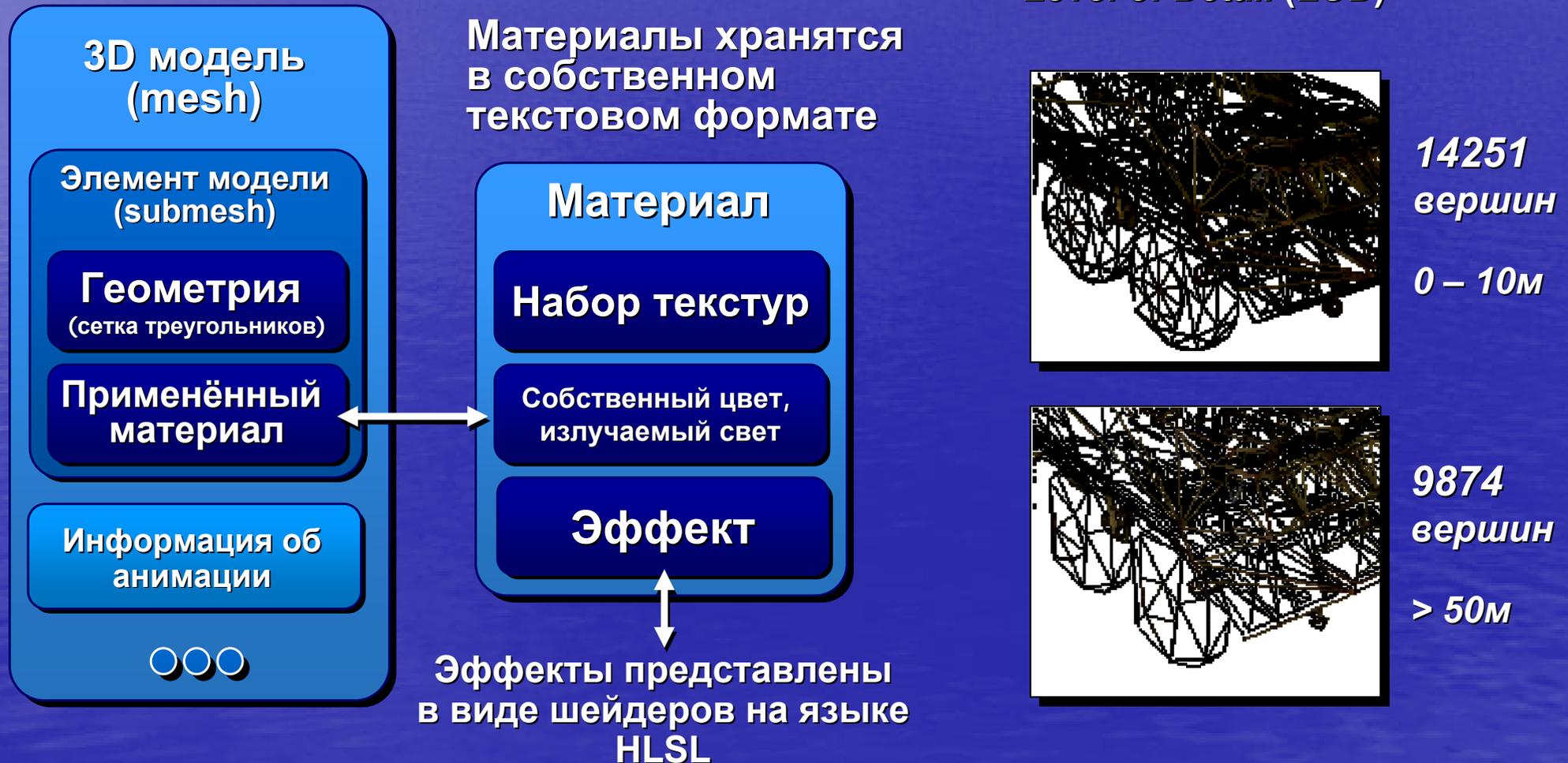


ГРАФИЧЕСКАЯ ПОДСИСТЕМА

Низкоуровневый графический API – Microsoft DirectX 9

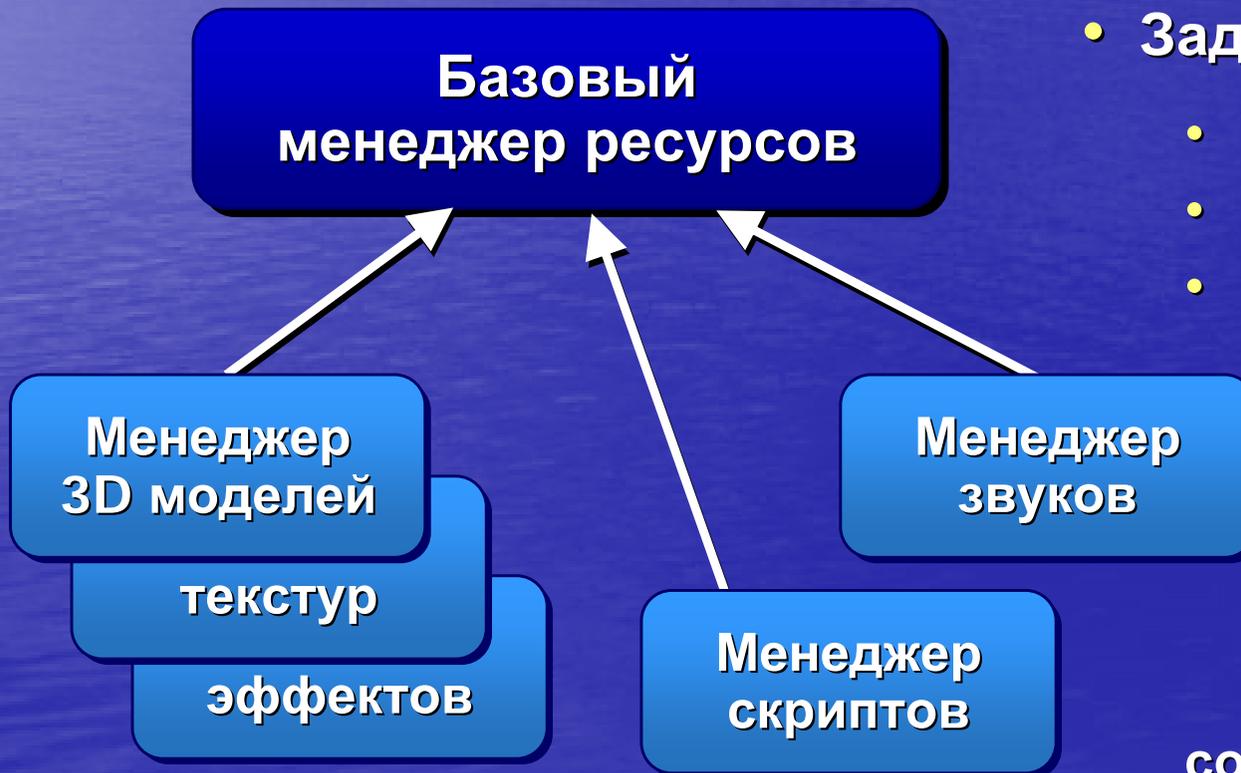
Движок работает с 3D моделями в бинарном формате DirectX

С целью повышения производительности используется технология Level of Detail (LOD)



УПРАВЛЕНИЕ РЕСУРСАМИ

- Ресурсы – любые внешние данные (модели, текстуры, звуки, скрипты, эффекты и т.п.)
- Для управления – менеджеры ресурсов, специфичные для каждого типа



- Задачи менеджера ресурсов:
 - Загрузка ресурсов
 - Освобождение ресурсов
 - Выдача ресурсов

Доступ к любому ресурсу предоставляется соответствующим менеджером

СКРИПТОВАЯ ПОДСИСТЕМА

Движок поддерживает скрипты на языке Lua

Причины выбора Lua:

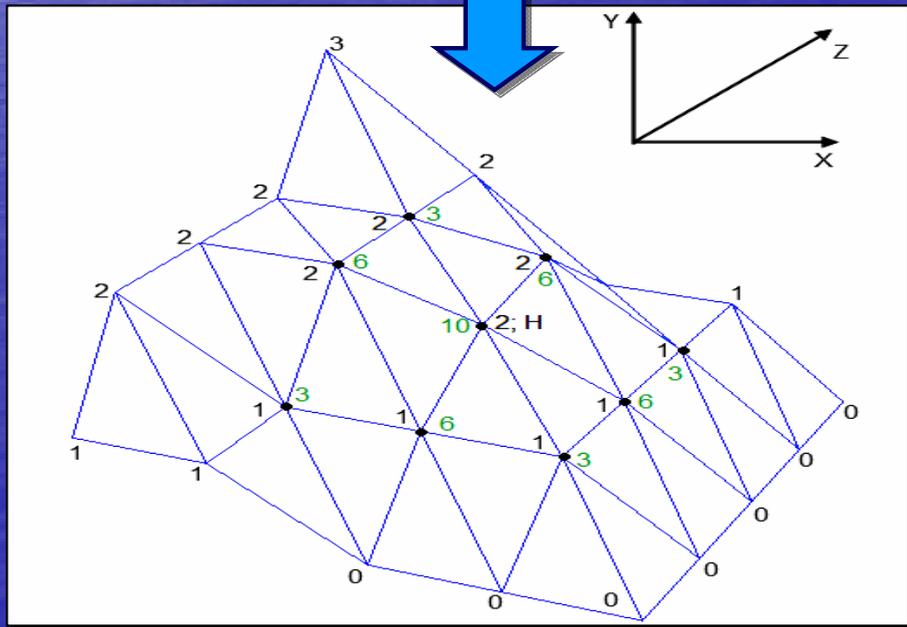
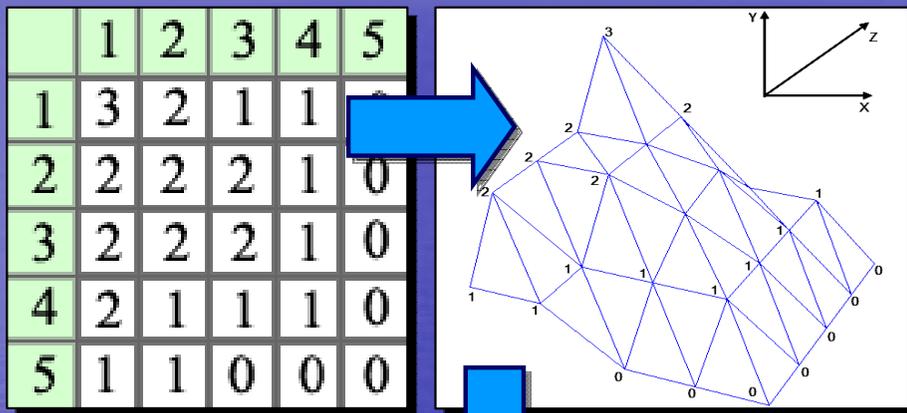
- Самый производительный скриптовый язык на сегодняшний день
- Обладает широким функционалом и возможностями
- Имеет большое количество пользователей и документации

При помощи скриптовой подсистемы реализуется поддержка технологий автоматического программирования



СОЗДАНИЕ СЕТКИ ЛАНДШАФТА

На основании данных из файла генерируются вершины сетки треугольников ландшафта



Сглаживание сетки ландшафта:

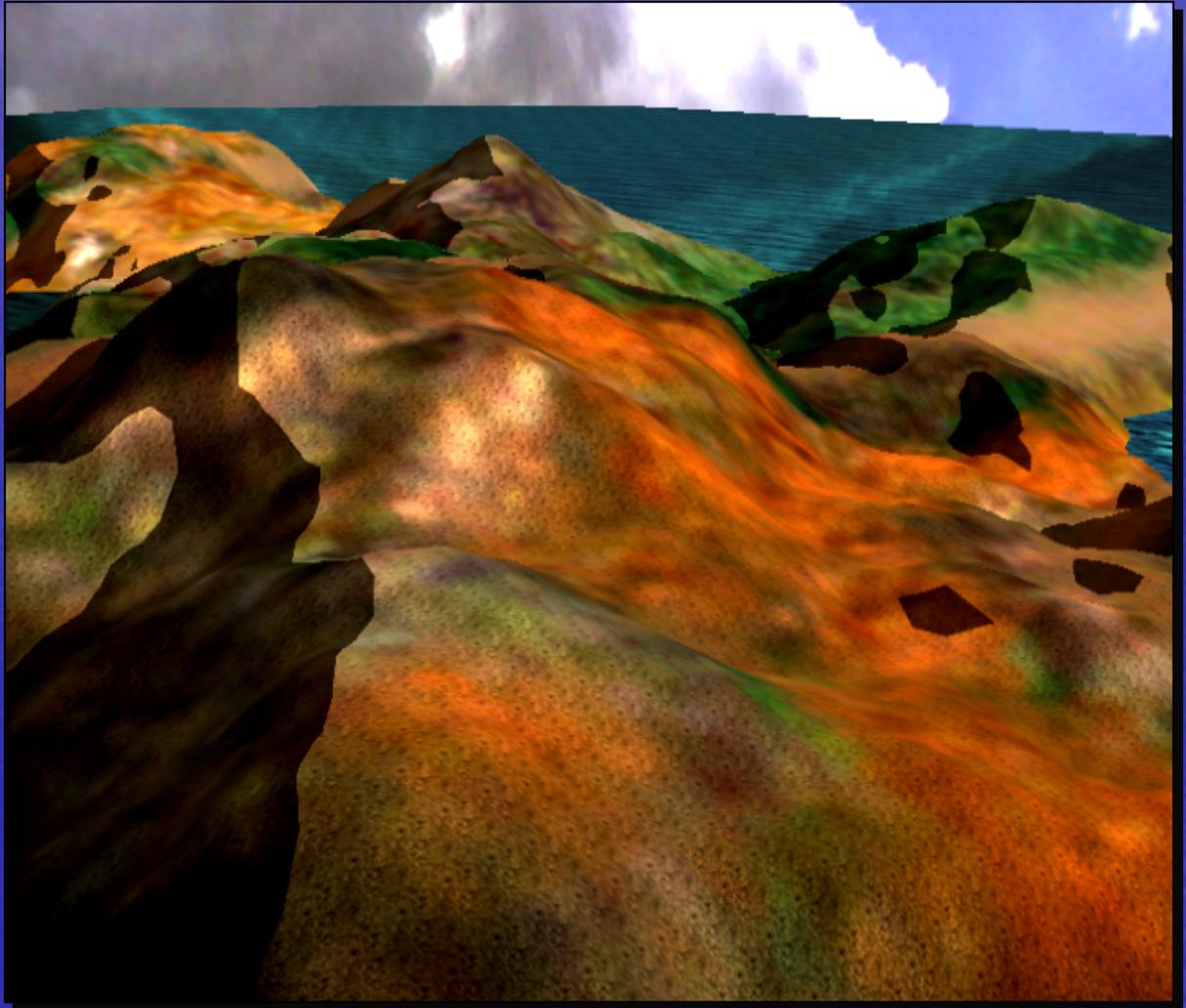
1. Присвоение весов (m_1, m_2, \dots) вершинам, соседним данной (высоты h_1, h_2, \dots).
2. Расчёт веса вершины по формуле

$$H = \frac{h_1 * m_1 + h_2 * m_2 + \dots}{m_1 + m_2 + \dots}$$

Зеленые числа – веса вершин,
черные – высоты

РАЗБИЕНИЕ ЛАНДШАФТА И ТЕКСТУРИРОВАНИЕ

- Разделение на части. Превращение каждой части в сетку треугольников
- Мультитекстурирование, с использованием глобальной и детальной текстур ($DstColor = 2 * GlobalColor * DetailColor$)
- Вершинное освещение
- Регистрация в физической подсистеме

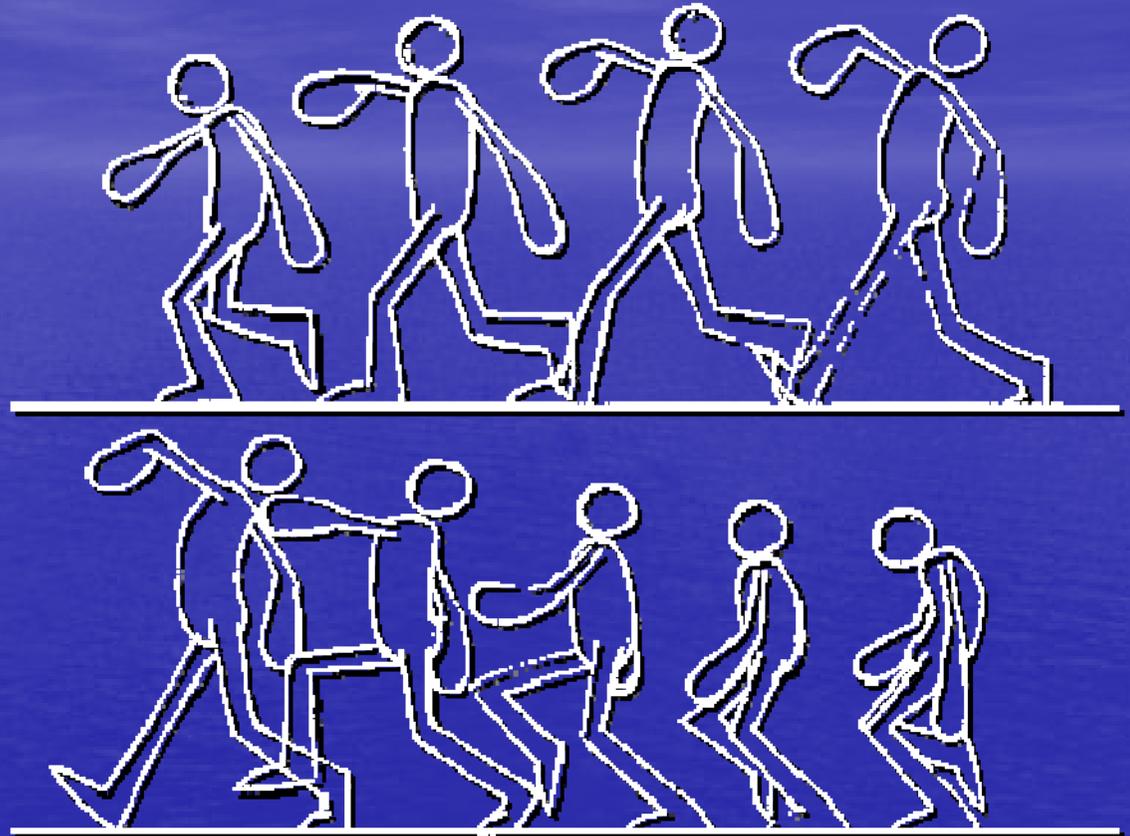


МИНУСЫ ФРЕЙМОВОЙ АНИМАЦИИ

≈24 кадра перехода,
10 положений

240 разных файлов
 $240 * 200 \text{ KB} = 48 \text{ MB}$
без сжатия

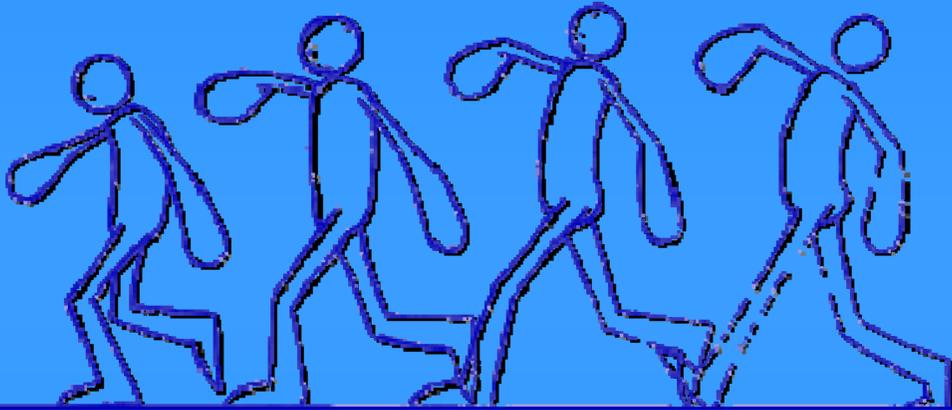
60 полных перерисовок,
около двух недель
разработки



Со сжатием: 1 персонаж = 2 MB.
Всего около 100 персонажей —
графика весит до 200 MB.
При смене внешнего вида персонажа 50%
кадров перерисовываются заново

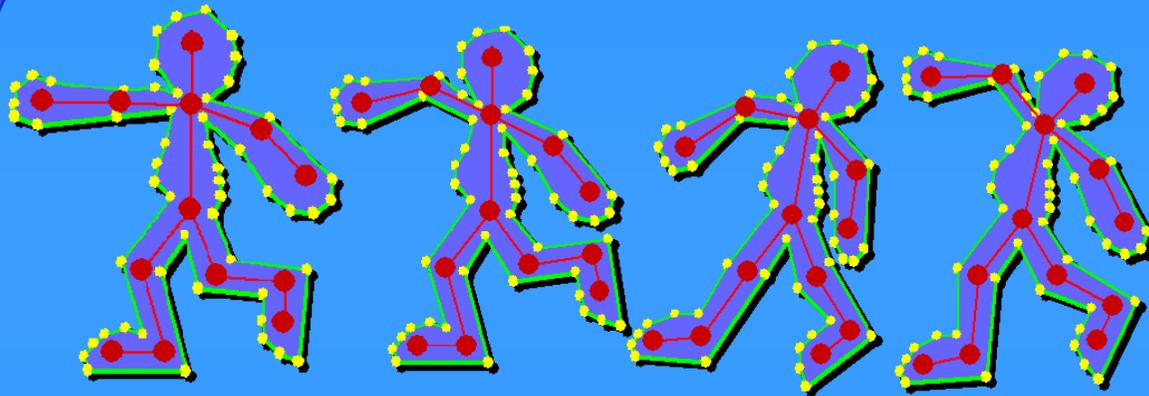
Абсолютная непригодность
для 3D графики по размеру
и времени создания

СКЕЛЕТНАЯ АНИМАЦИЯ



В файле содержится
каждый кадр анимации

Фреймовая анимация



В файле содержатся
1. Форма модели
2. Кости модели
3. Наборы положения костей
4. Текстура модели для
одного из кадров

Скелетная анимация

СКЕЛЕТНАЯ АНИМАЦИЯ: АНАТОМИЯ

Контур (lines)

В файле хранятся :
 i, j — *индексы вершин*

Суставы (joints)

В файле хранятся :
 x, y - координаты

Вершины (vertex)

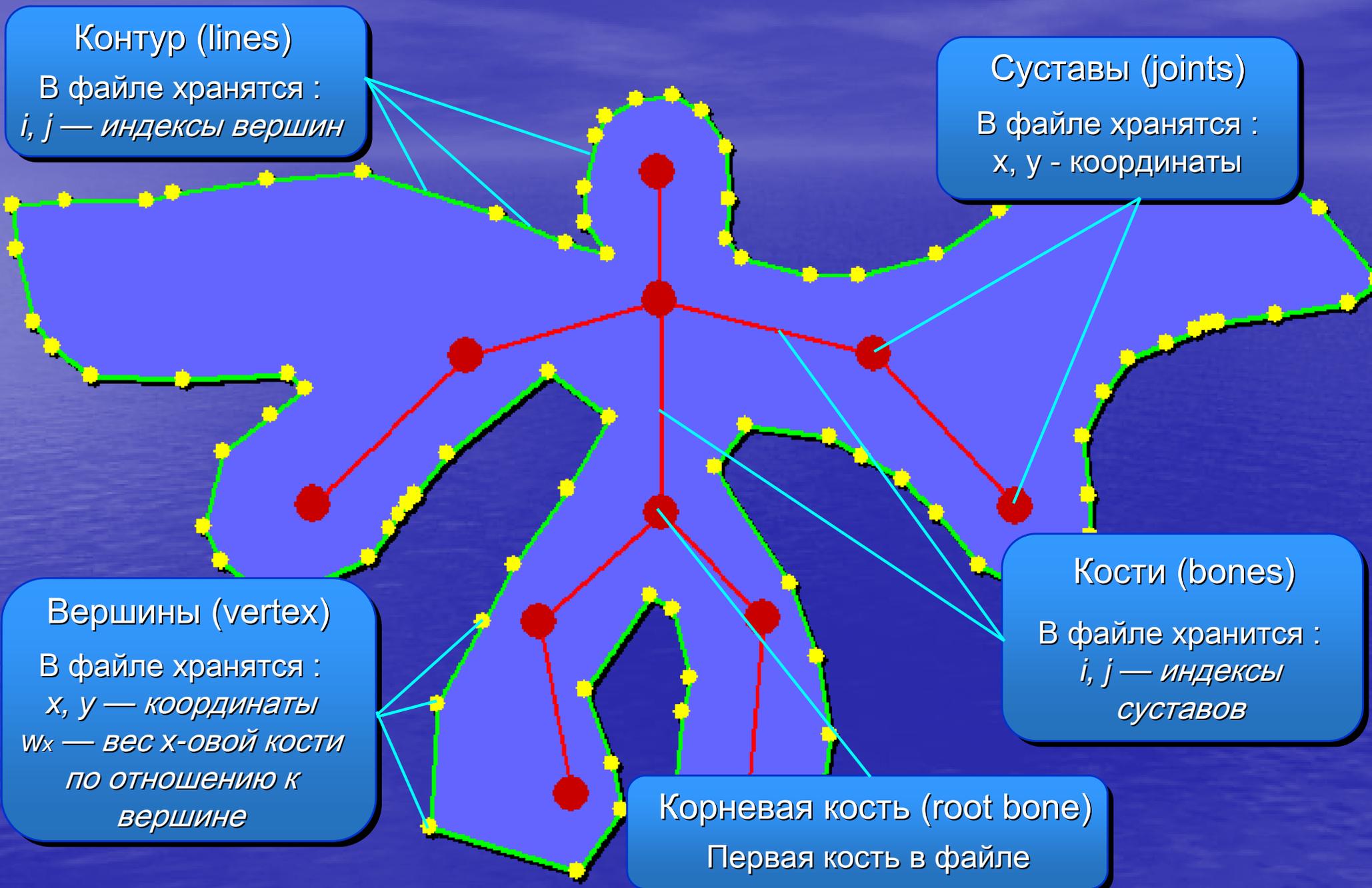
В файле хранятся :
 x, y — *координаты*
 w_x — *вес x-овой кости по отношению к вершине*

Кости (bones)

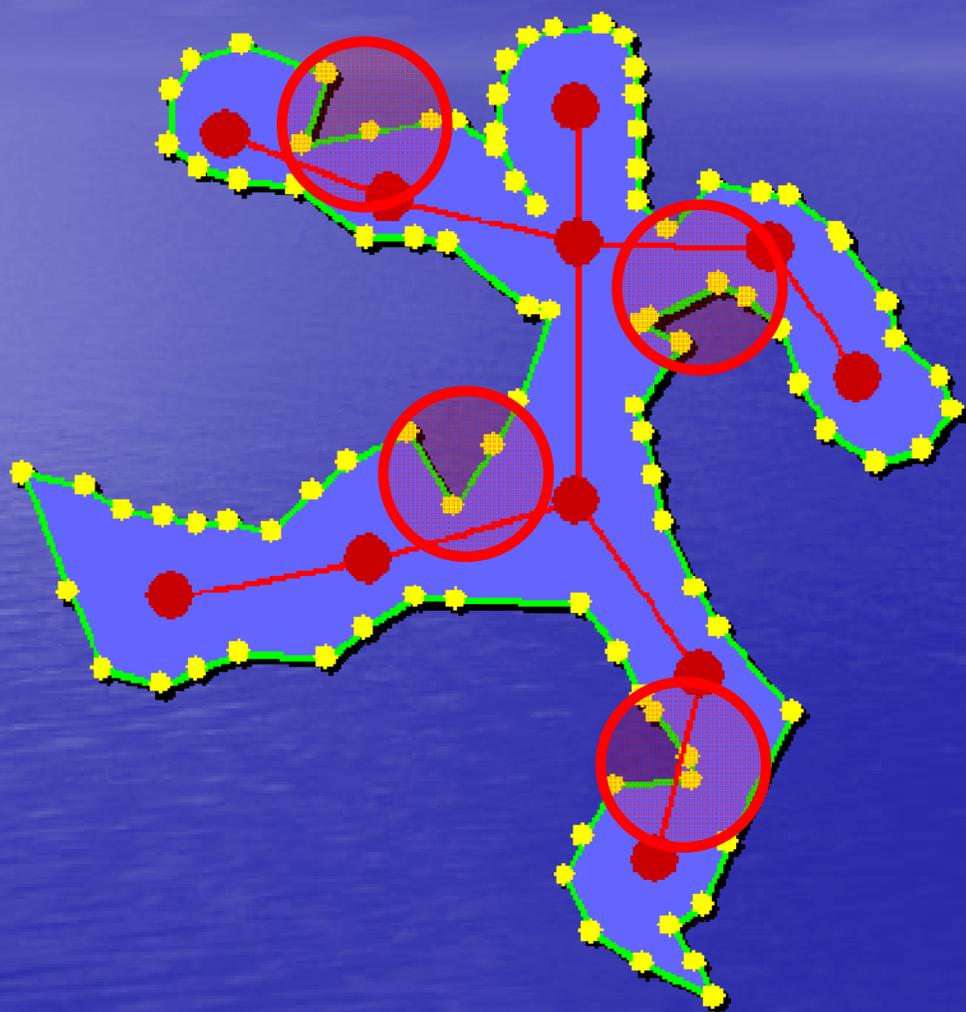
В файле хранится :
 i, j — *индексы суставов*

Корневая кость (root bone)

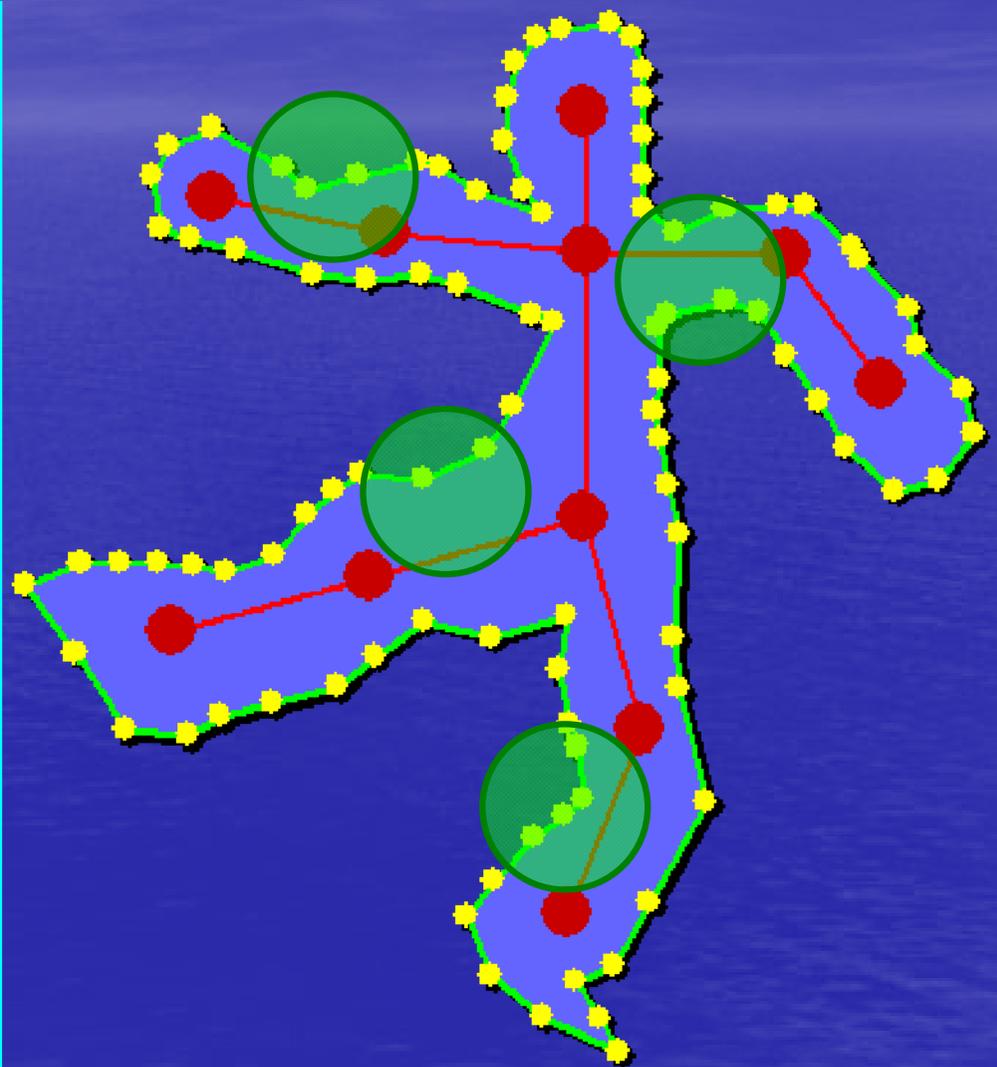
Первая кость в файле



ЗАЧЕМ НУЖЕН СКИННИНГ?



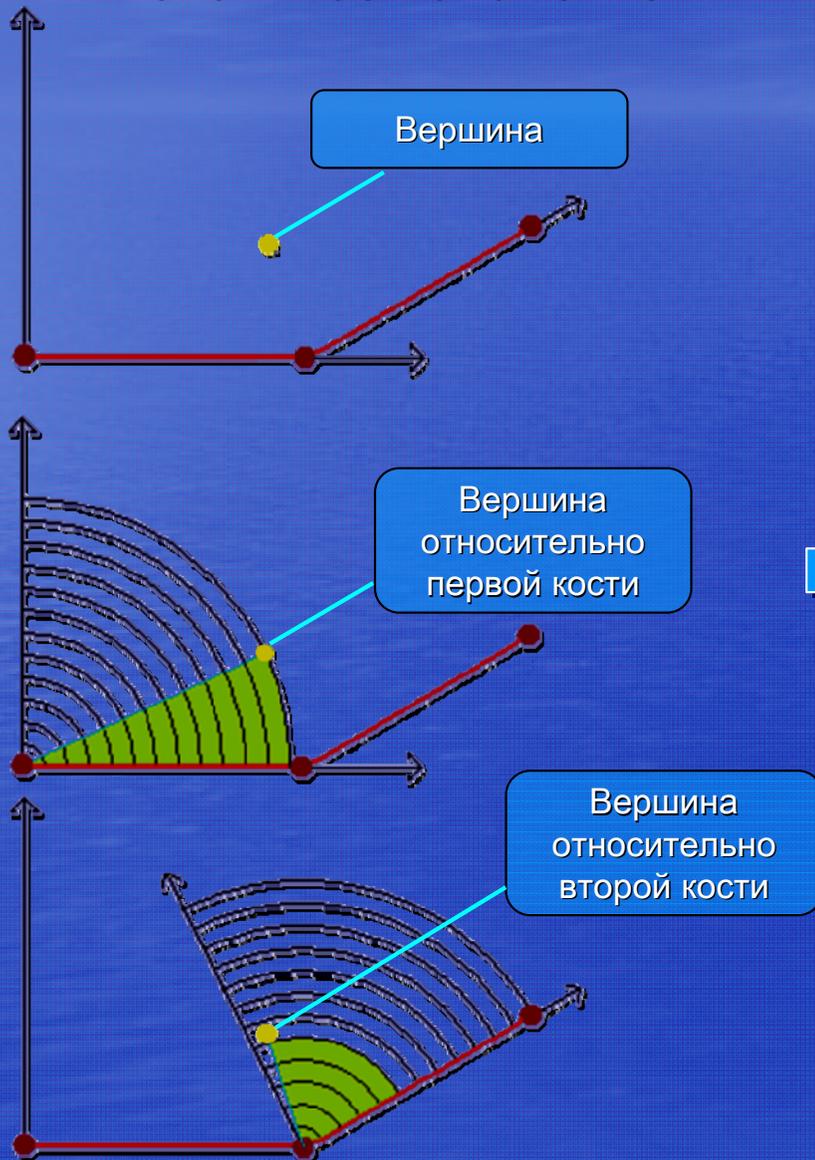
Без скиннинга



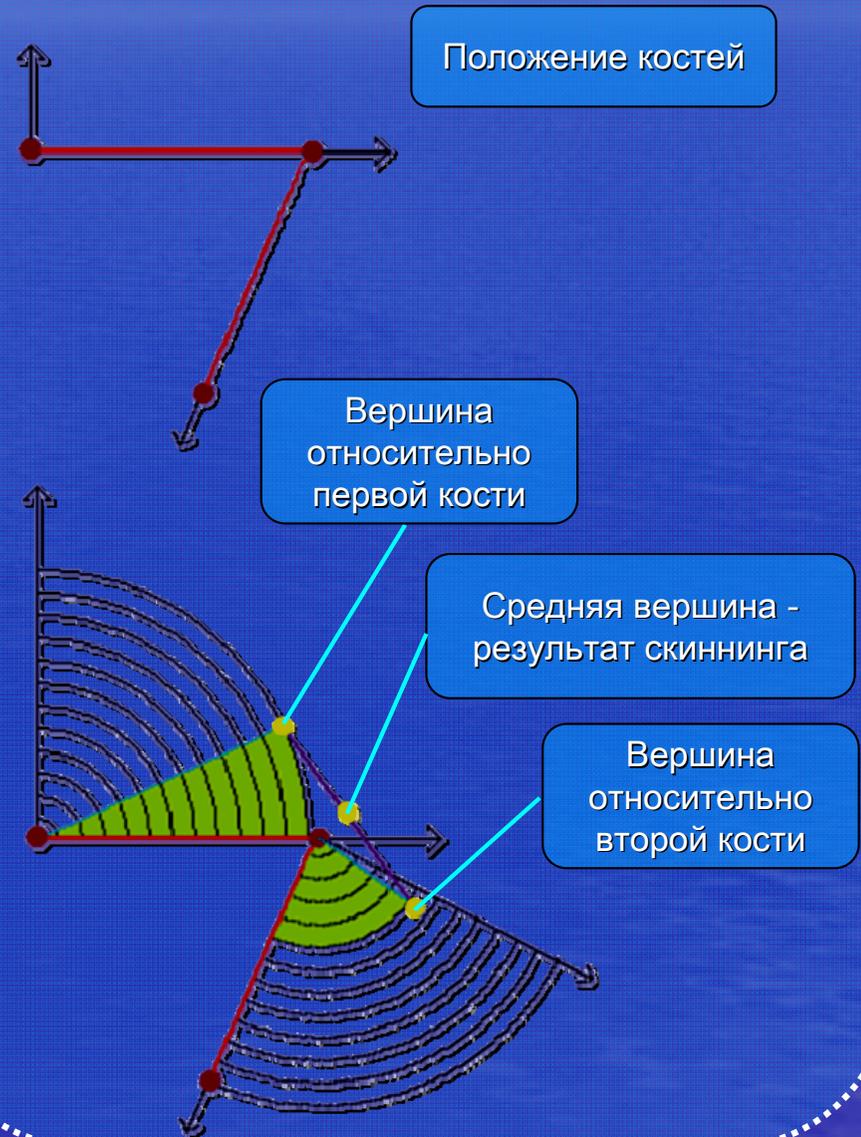
Со скиннингом

СКИННИНГ

Начальное положение



Конечное положение



СКИННИНГ: АВТОРАЗВЕСОВКА

Начало

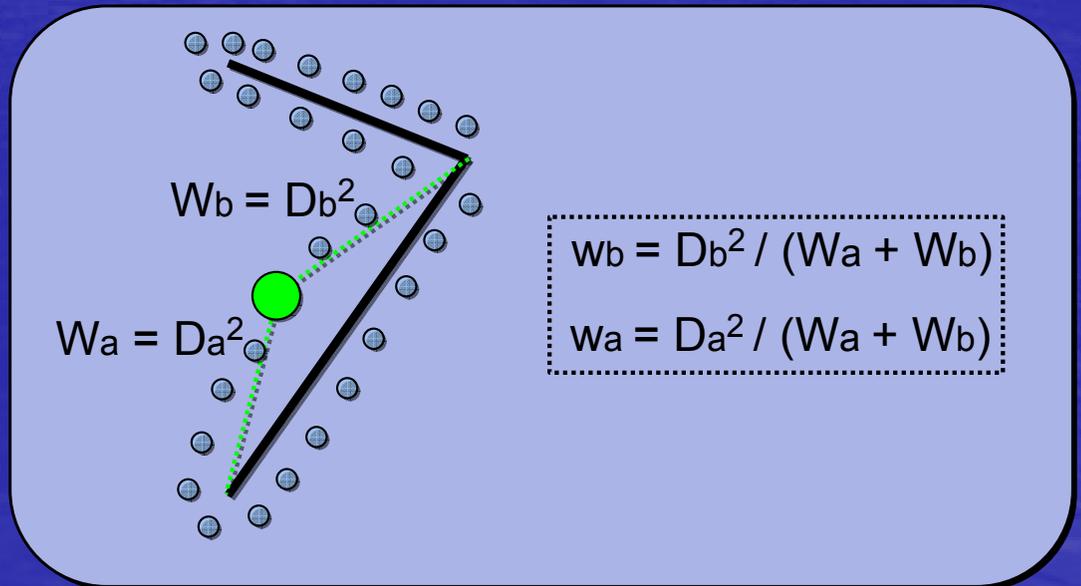
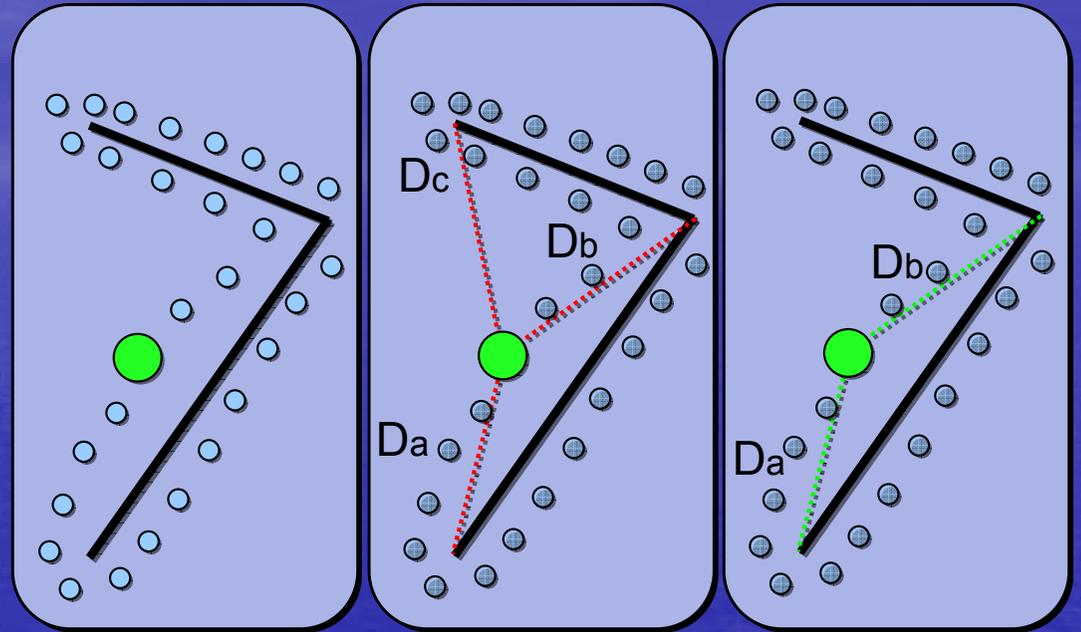
Вычислить расстояние от вершины до всех суставов (joints)

Выбрать N самых близких суставов

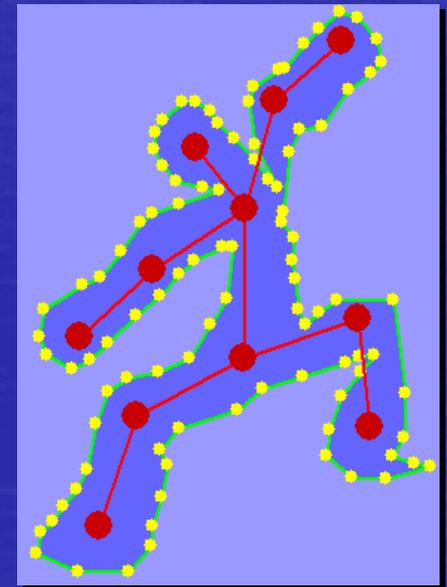
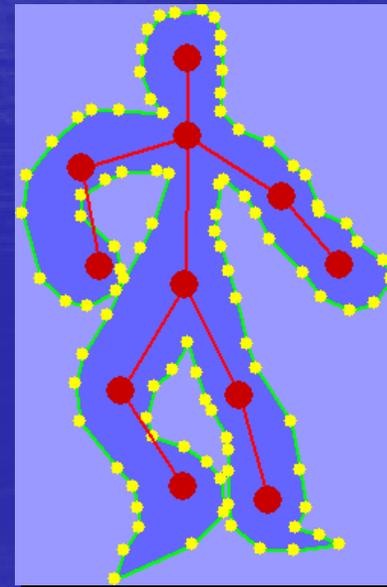
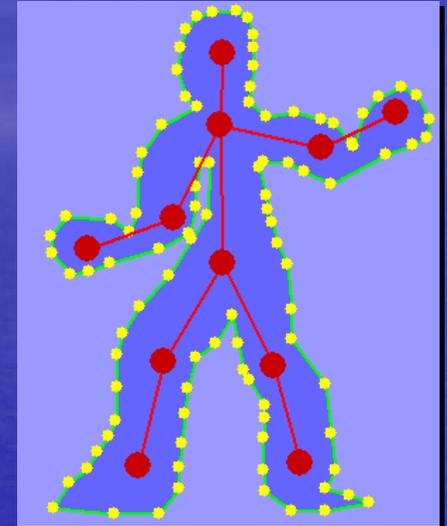
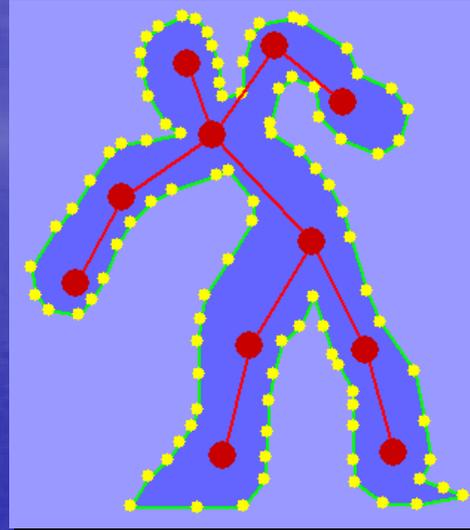
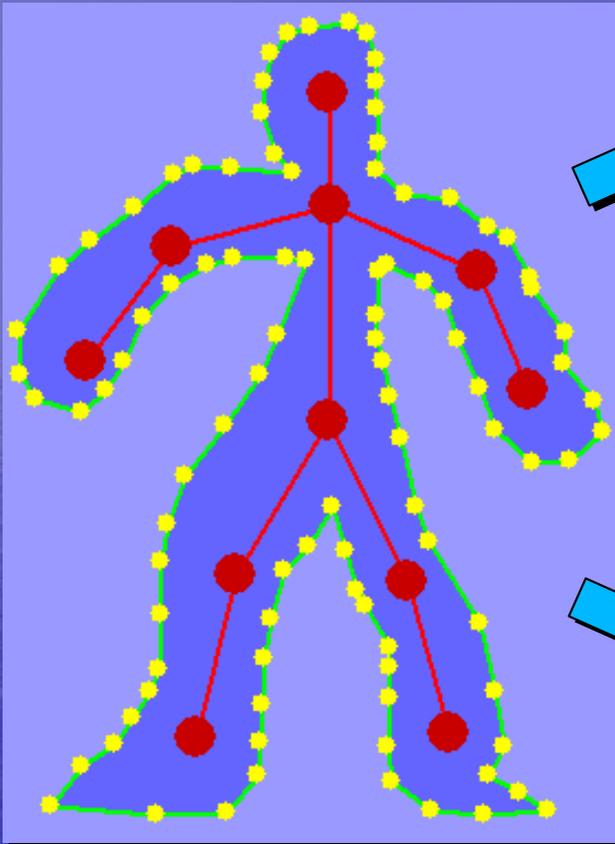
Вычислить вес каждой кости по формуле $1 / (\text{dist}^2)$

Привести сумму весов к 1, не изменяя их соотношений

Конец



ПРИМЕР СКИННИНГА



МОДУЛЬ СКЕЛЕТНОЙ АНИМАЦИИ

Редактор анимации

Формат анимационных файлов

Готовых форматов в 2D нет
Для 3D нужна система загрузки

Алгоритм авторазвесовки

Определяет степень влияния каждой кости на каждую вершину

Движок анимации

Алгоритм скиннинга

Отвечает за нахождение координат вершин по положению костей

Расширенный графический модуль

Перенос вычислений на видеокарту.
Модуль не обязателен



Вывод графики

Взаимодействие с движком

Расчёт физики

СКЕЛЕТНАЯ АНИМАЦИЯ В 3D



Архитектура автоматного боя онлайн

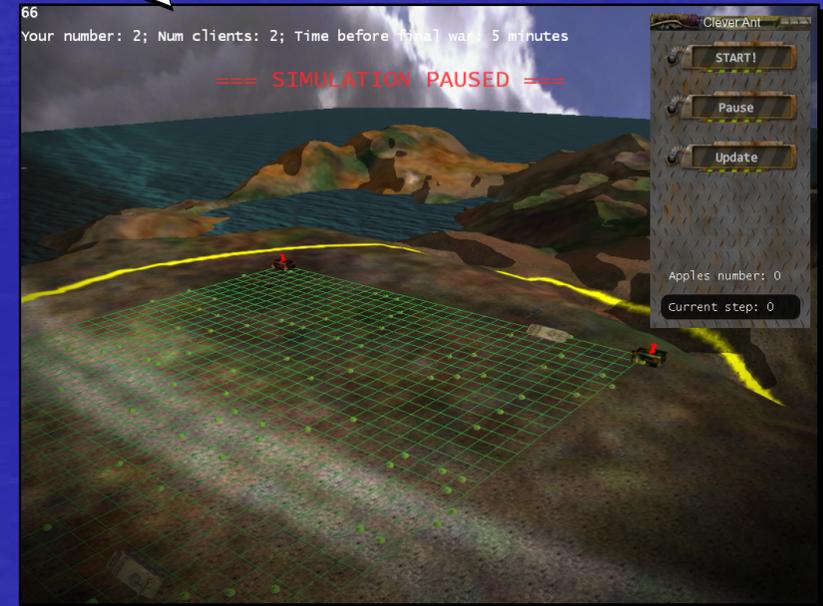
Сервер

```
XEngine log window
XNetServer: Creating clients structures...
XScript: loading script from file 'Scripts/_Automata/Intellect/AutomataLog.lua'
WARNING: Resource loading before full engine initialization!!!
XScript: loaded successfully
XScript: finished
XScript: loading script from file 'Scripts/_Automata/Intellect/AutomataCore.lua'
WARNING: Resource loading before full engine initialization!!!
XScript: loaded successfully
XScript: finished
XScript: loading script from file 'Scripts/_Automata/Intellect/Automata.lua'
WARNING: Resource loading before full engine initialization!!!
XScript: loaded successfully
XScript: finished
Enter map number (Deathville - 1), (Prototype - 2)
2
Entering map #2 (Prototype)
Enter war time (in minutes)
5
XCore: new module attached; address = '269514120'
XCore: new module attached; address = '269520440'
XCore: new module attached; address = '269514060'
XCore: ===== core started =====
XCore: total modules number: 3
```

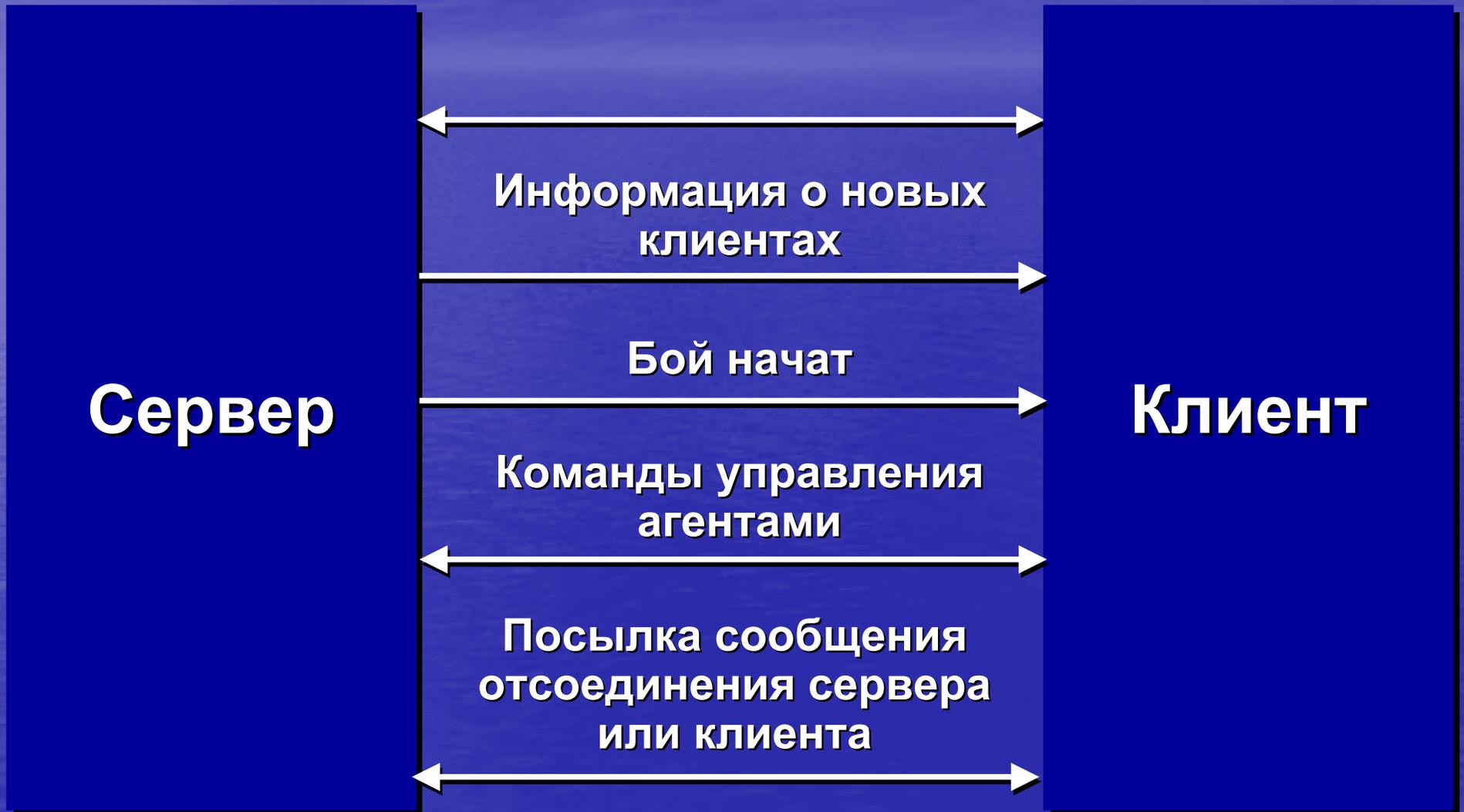
Клиент



Клиент



Обмен данными в сети



РЕЗУЛЬТАТЫ

**Создана первая версия движка
для платформы автоматного
моделирования**

Движок для платформы поддерживает:

- **Загрузку и отображение трехмерного мира**
- **Воспроизведение различных звуков и видеороликов**
- **Реалистичную симуляцию взаимодействия объектов**
- **Внутриигровую систему графического интерфейса**