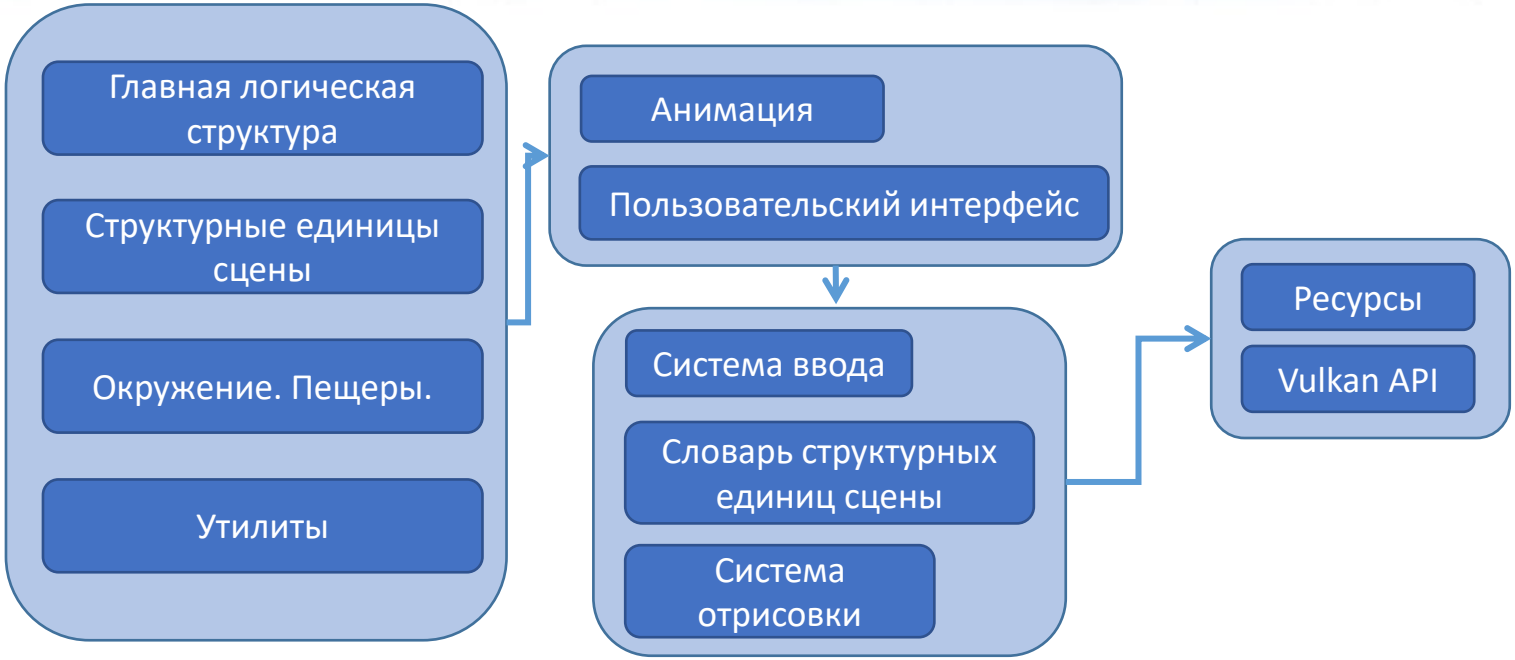
Разработка программы построения и визуализации пещерных систем

Худяков Тимофей 09-4 класс
Ильясов Александр 10-3 класс
Витман Илья 11-2 класс
Пашков Иван 11-5 класс
Топунов Антон 11-6 класс
Фисунова Лидия 11-6 класс

Научный руководитель: Галинский В.А., руководитель группы компьютерной графики,
преподаватель информатики и программирования СПб губернаторского физико-математического лицея № 30

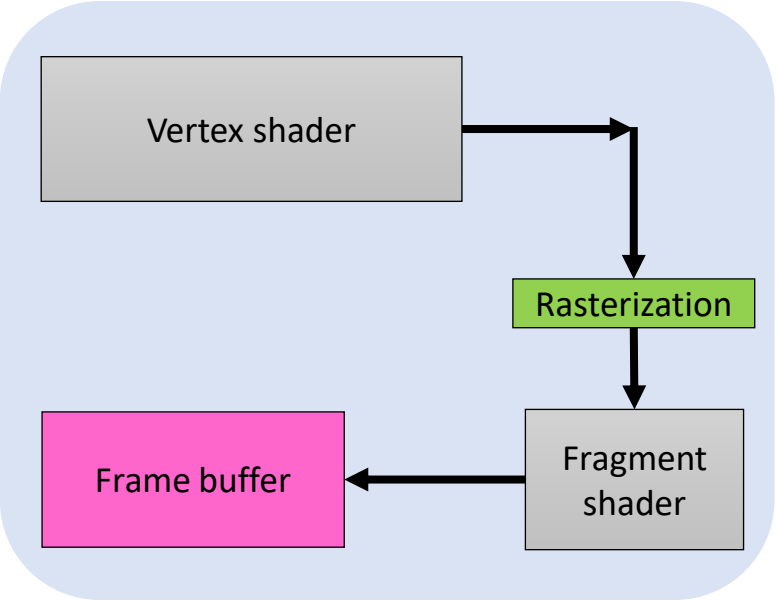
Computer Graphics Support Group
Санкт-Петербургский губернаторский физико-математический лицей № 30

Структура проекта

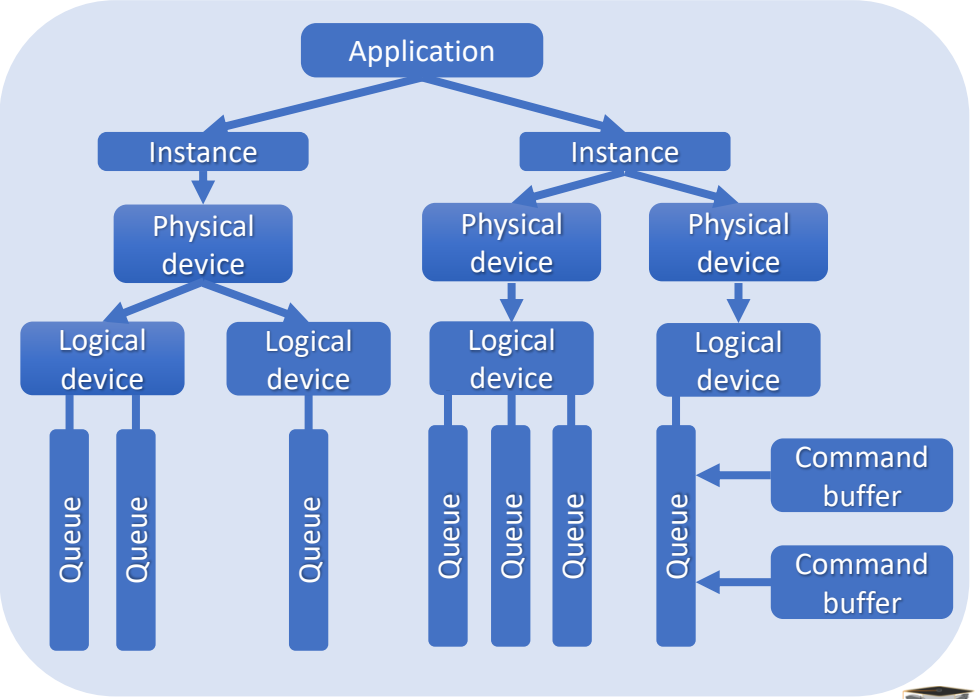


Vulkan API

Базовый конвейер вывода Vulkan API
(На уровне GPU)



Общая блок-схема работы с Vulkan API

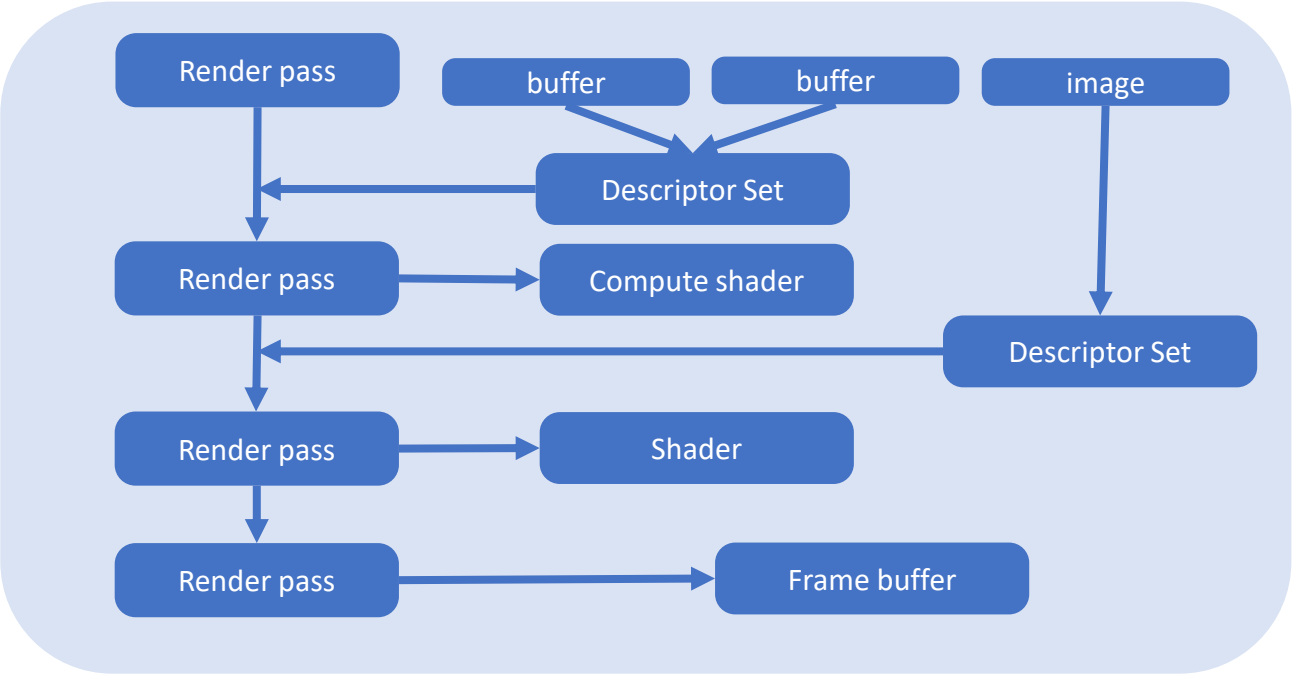


Vulkan resources

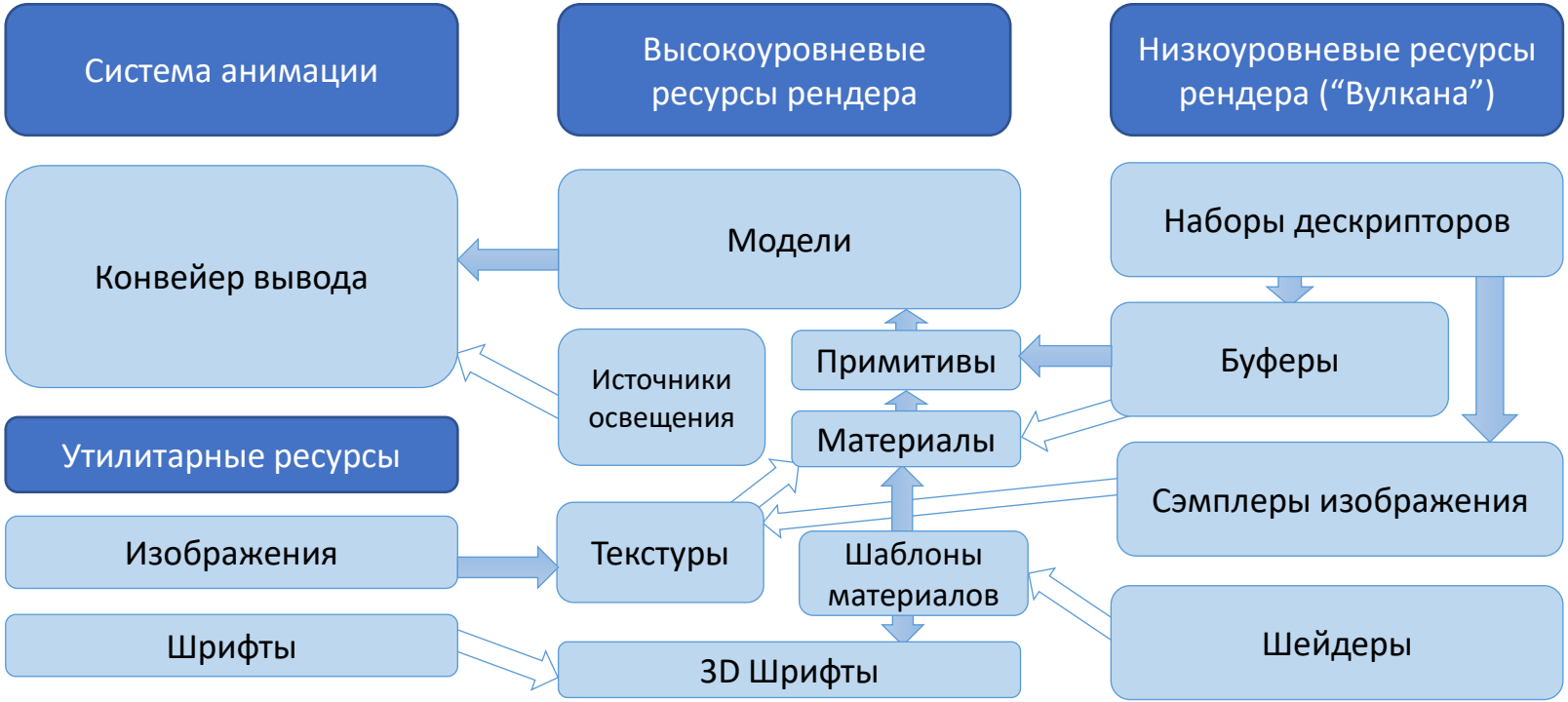
Схема использования низкоуровневых ресурсов
во время конвейера вывода

Состав низкоуровневых ресурсов:

- Vulkan buffers;
- Vulkan frame buffers;
- Vulkan images;
- Vulkan descriptor sets;
- Vulkan render passes;
- Vulkan shaders;
- Vulkan compute shaders;



Ресурсы


Система анимации
Высокоуровневые ресурсы рендера
Низкоуровневые ресурсы рендера (“Вулкана”)
Конвейер вывода
Модели
Наборы дескрипторов
Источники освещения
Примитивы
Буферы
Материалы
Утилитарные ресурсы
Сэмплеры изображения
Изображения
Текстуры
Шаблоны материалов
Шрифты
3D Шрифты
Шейдеры

Ресурсы. Шаблоны материалов

Схема ручного создания графического конвейера

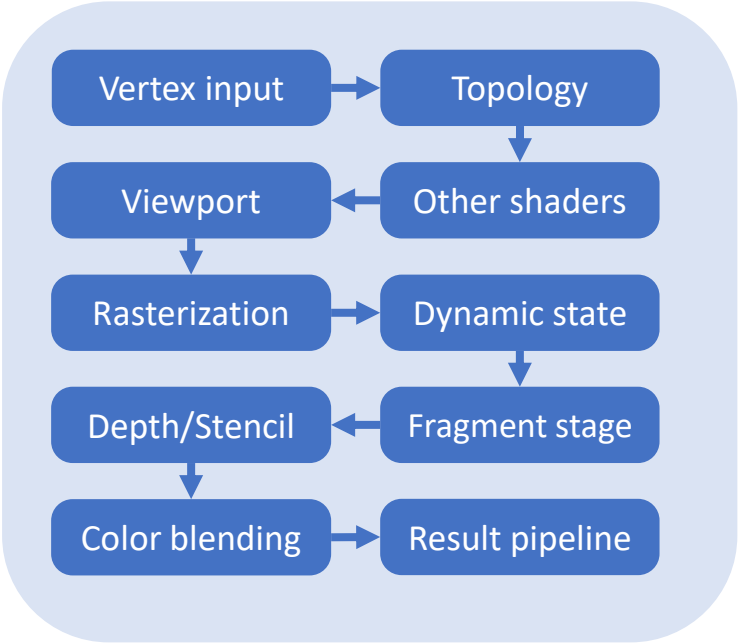
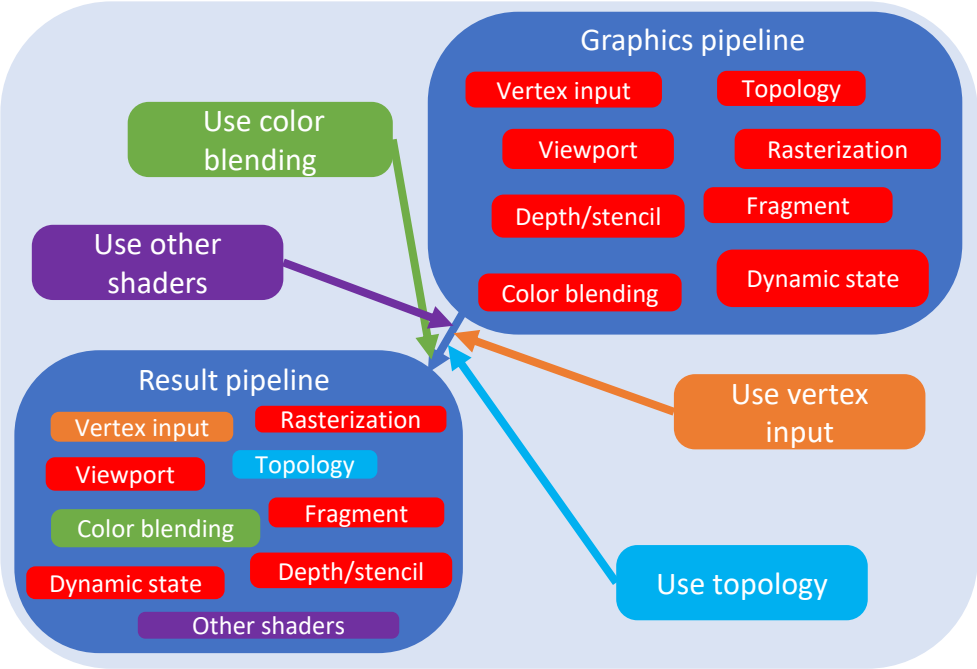
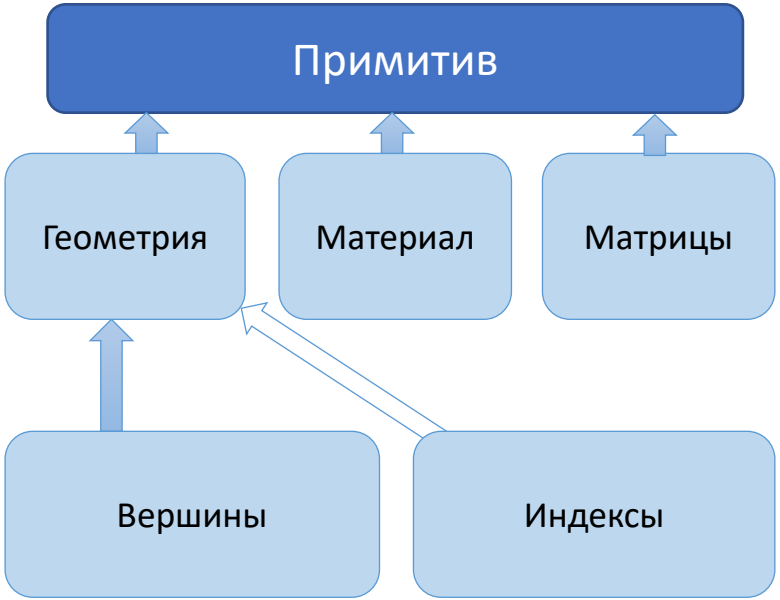
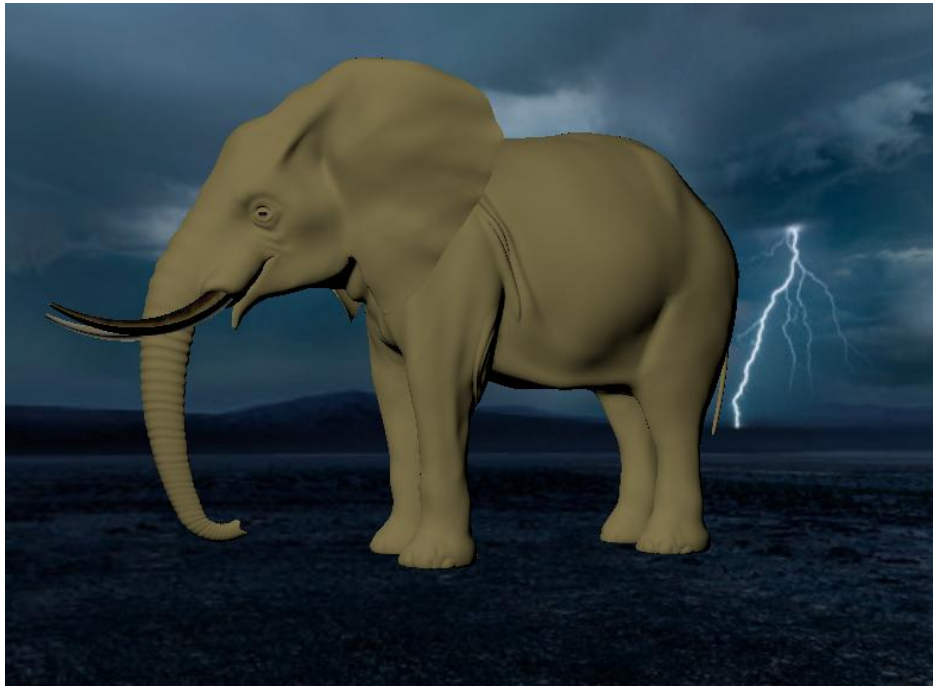


Схема нашего создания графического конвейера

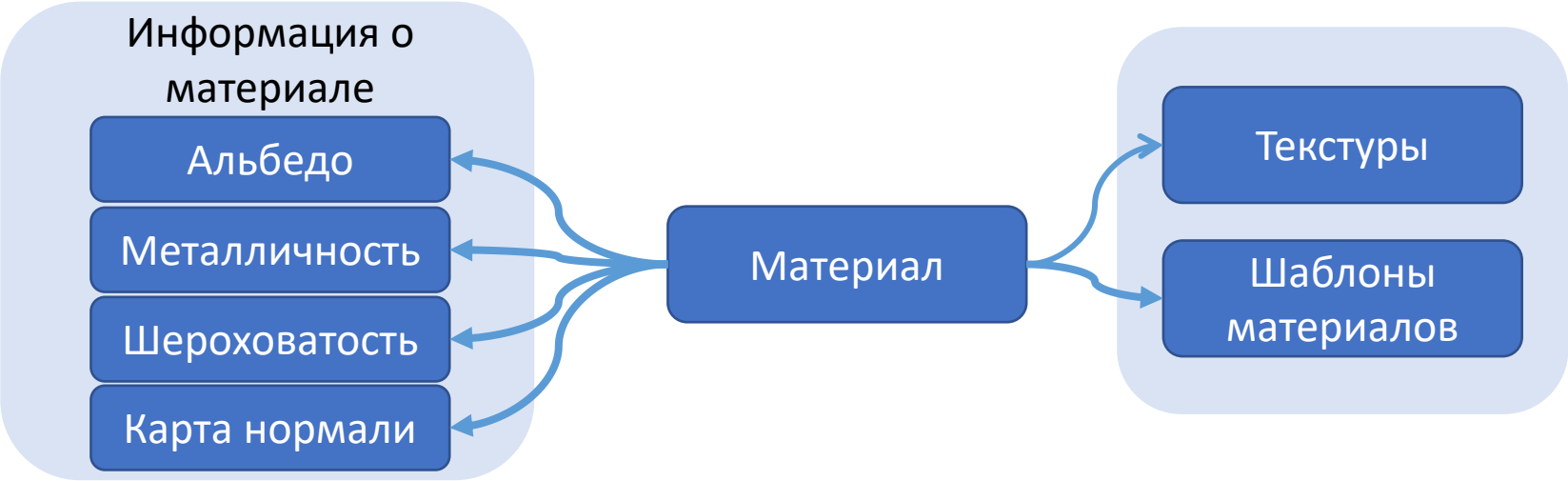


Ресурсы. Примитивы





Ресурсы. Материалы



Ресурсы. Текстуры

Cube-текстуры, накладываются на небо

2D текстуры (Albedo коэффициент и карта нормалей), накладываются на объект

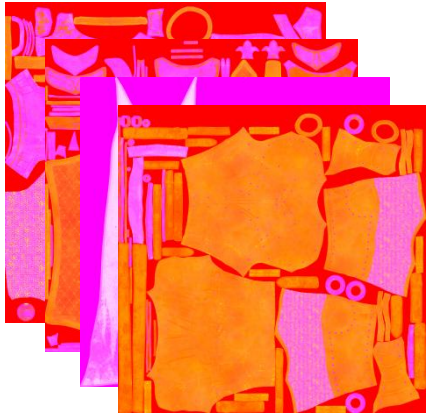
Ресурсы. Текстуры. Альбедо

Альбедо текстуры
(текстуры, использующиеся для отображения цвета самой модели)

Ресурсы. Текстуры. Карта нормалей

Текстура карты нормалей (используется для рельефного текстурирования)

Ресурсы. Текстуры. Металличность и шероховатость



Текстуры металичности и шероховатости (Металличность используется для определения металличности поверхности. Шероховатость используется для определения гладкости поверхности)

Пример металличности

Пример шероховатости

Ресурсы. Модели освещения

PBR

плюсы

- Честное освещение, основанное на физике
- Повседневно используется в 3d-графике

минусы

- Сложно написать правильно работающую модель

Phong

плюсы

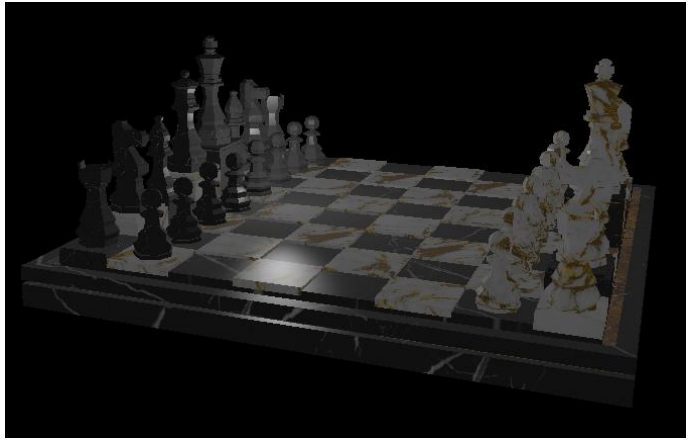
- Легко написать рабочую версию
- Быстрая обработка

минусы

- Устаревшая модель освещения
- Не основана на физике

Ресурсы. Модели освещения. Сравнение Phong и PBR

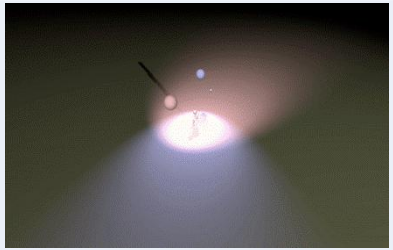
Phong



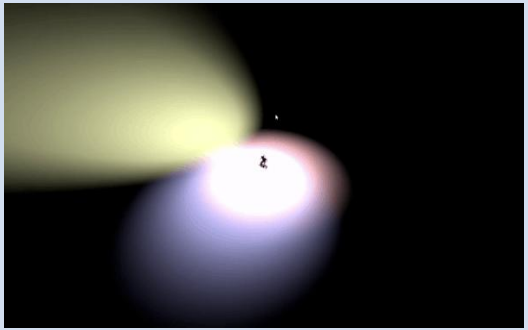
PBR

Ресурсы. Источники света. Тени

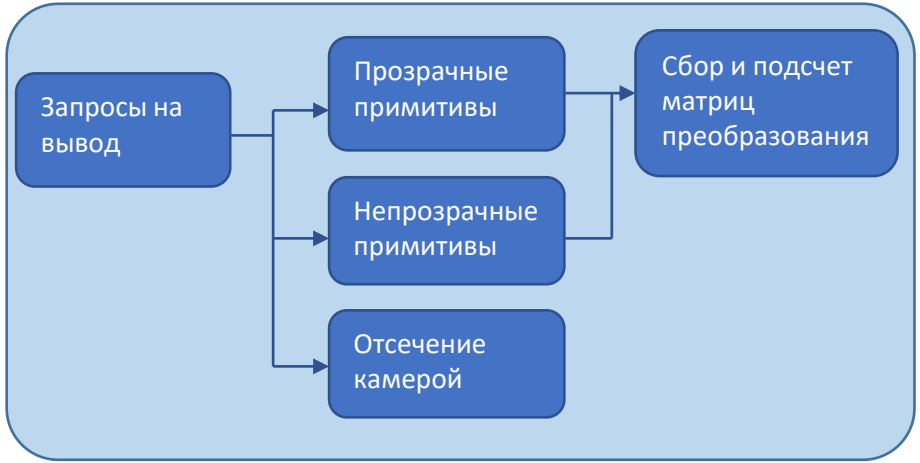
Ресурсы. Источники света. Тени.

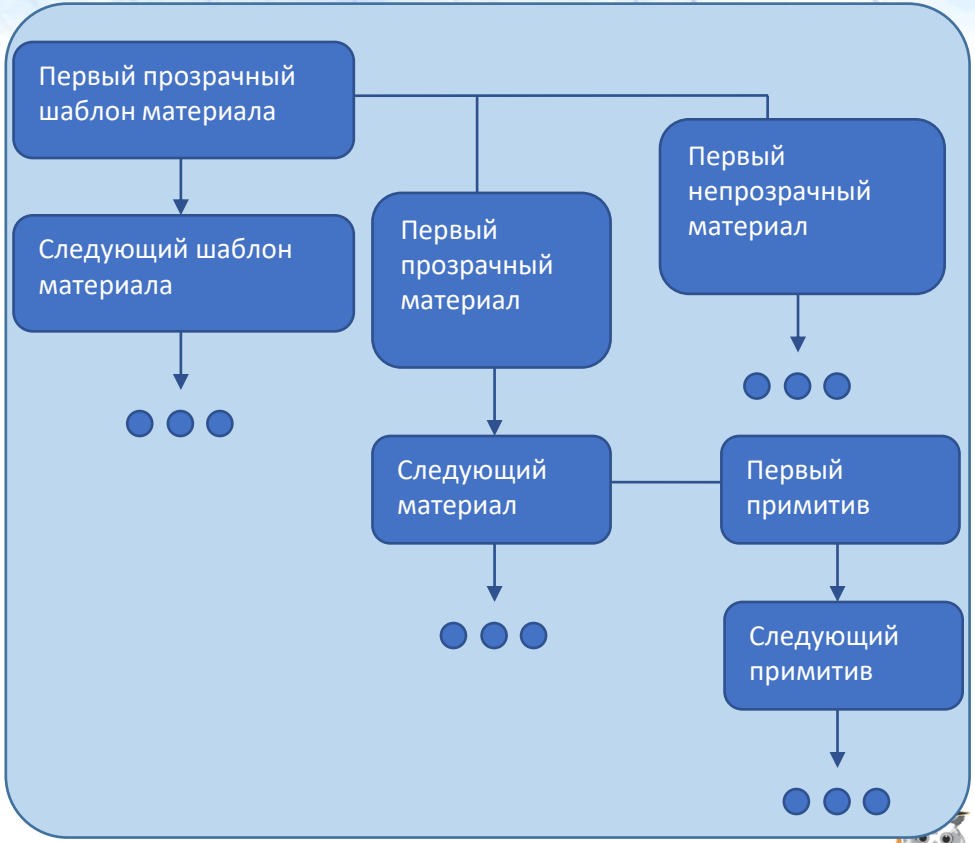
Тип источника освещения	Пример использования
Точечный источник	
Прожектор	

Ресурсы. Источники света. Тени.

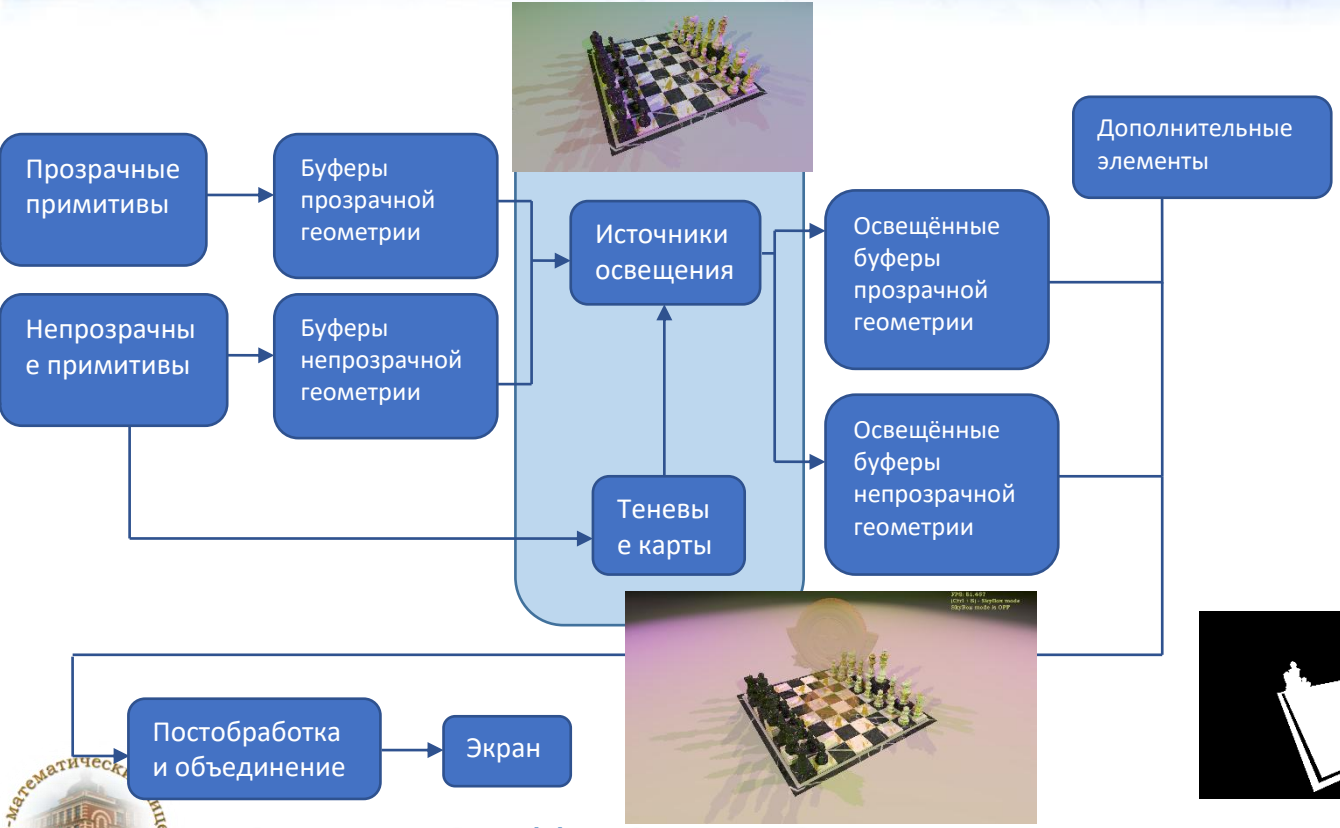
Тип источника освещения	Пример использования
Направленный	

Render. Графический конвейер

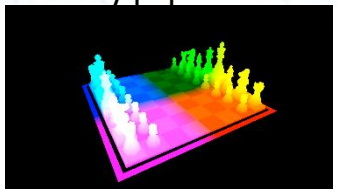


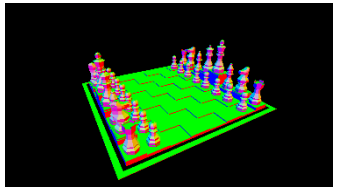


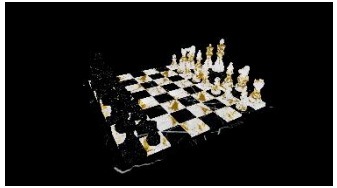
Render. Графический конвейер

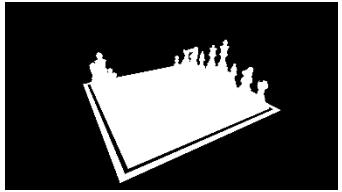


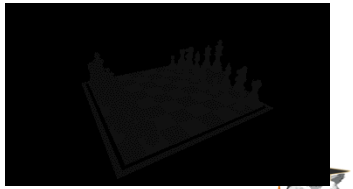
Геометрические буферы



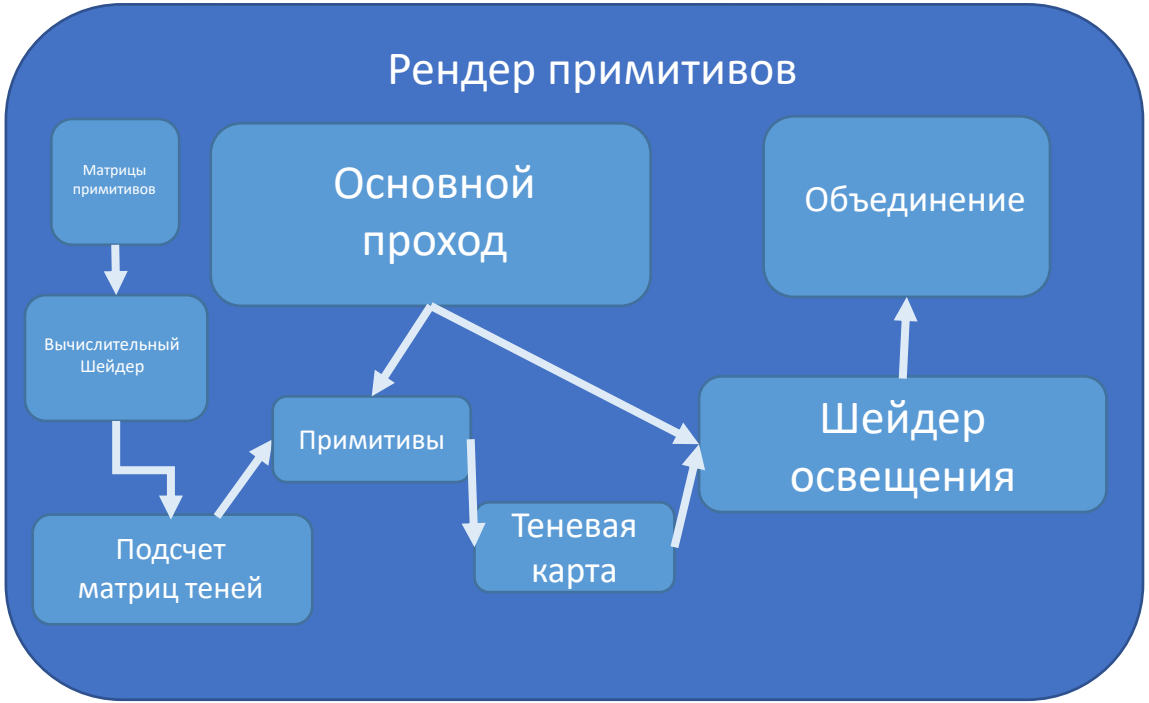








Тени


Рендер примитивов
Матрицы примитивов
Основной проход
Объединение
Вычислительный Шейдер
Примитивы
Шейдер освещения
Теневая карта
Подсчет матриц теней

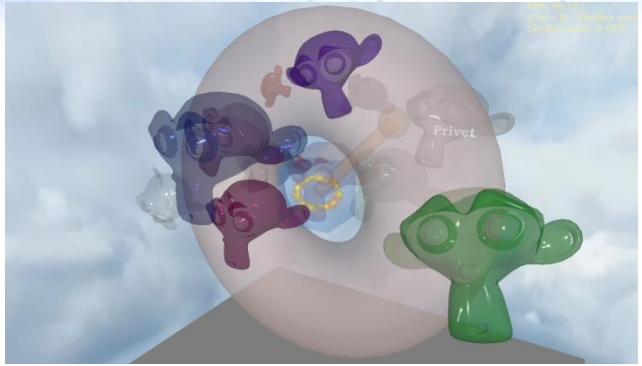
Pipeline. OIT

Прозрачный треугольник

Предварительное отсечение

Запись в буфер глубины и в геометрические буфера

Добавление данных об освещении в связный список пикселя



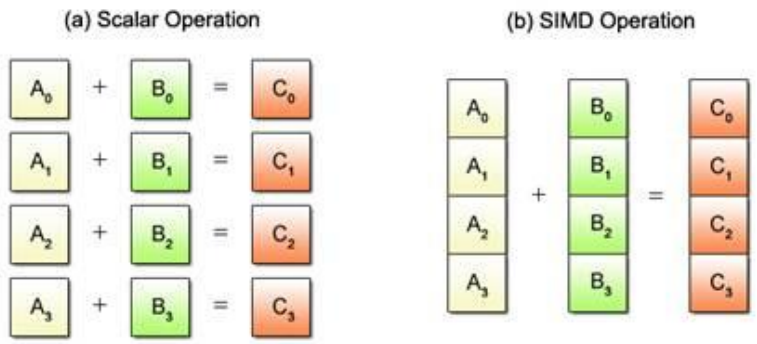
Изображение с индексами начал списков

Цвет
Коэффициенты освещения

Следующий элемент
Глубина
Перспективный делитель

Private

SIMD-оптимизация вычислений



Сложение векторов отдельно по скалярам:

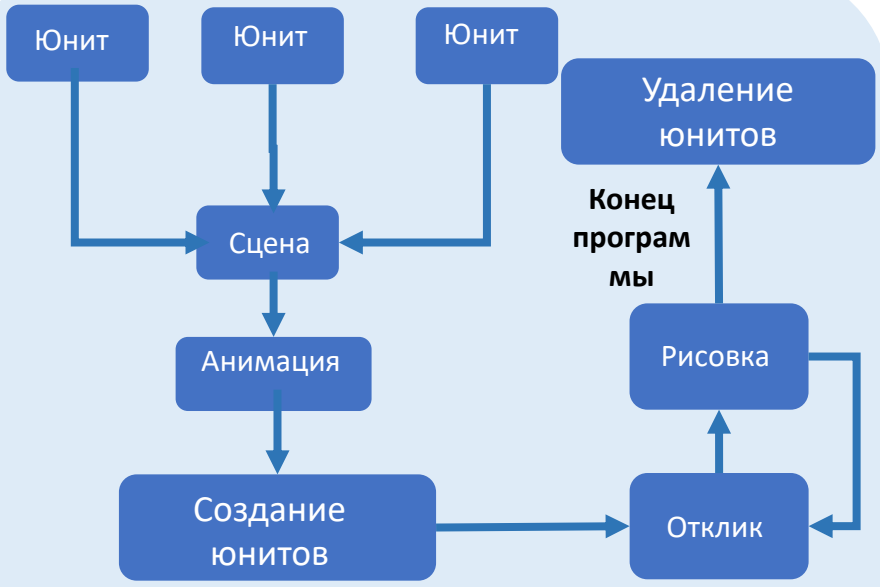
```
V3.X = V1.X + V2.X;
V3.Y = V1.Y + V2.Y;
V3.Z = V1.Z + V2.Z;
V3.W = V1.W + V2.W;
```

Векторизация сложения при помощи SIMD-инструкций:

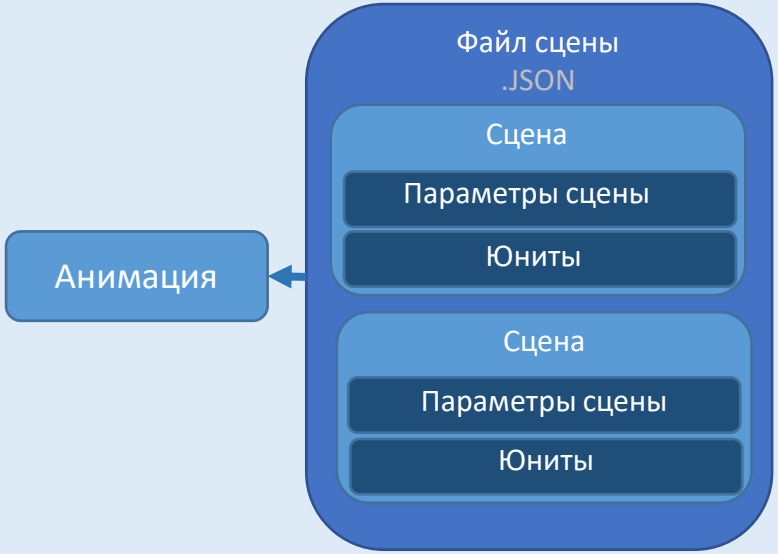
```
_mm_store_ps(&V3.X,
             _mm_add_ps(_mm_load_ps(&V1.X),
                        _mm_load_ps(&V2.X)));
```

Использованы инструкций из расширений: SSE family (SSE, SSE2, SSE3, SSE4), AVX family (AVX, AVX2)

Анимация. Сцена

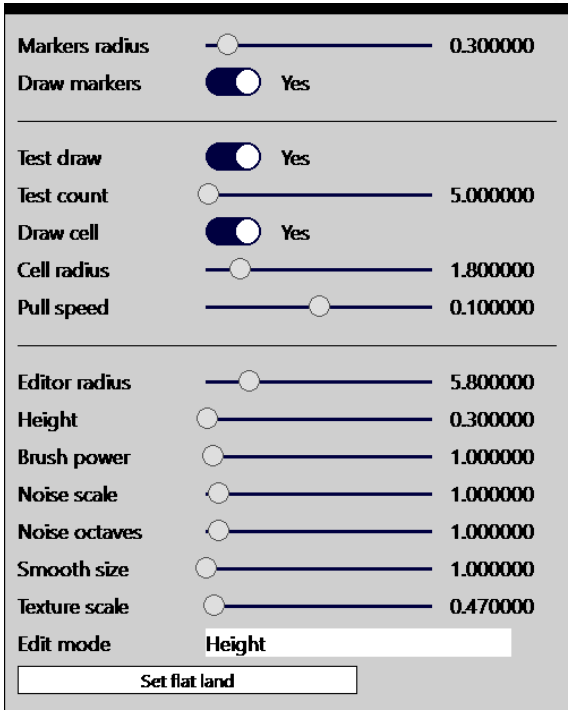


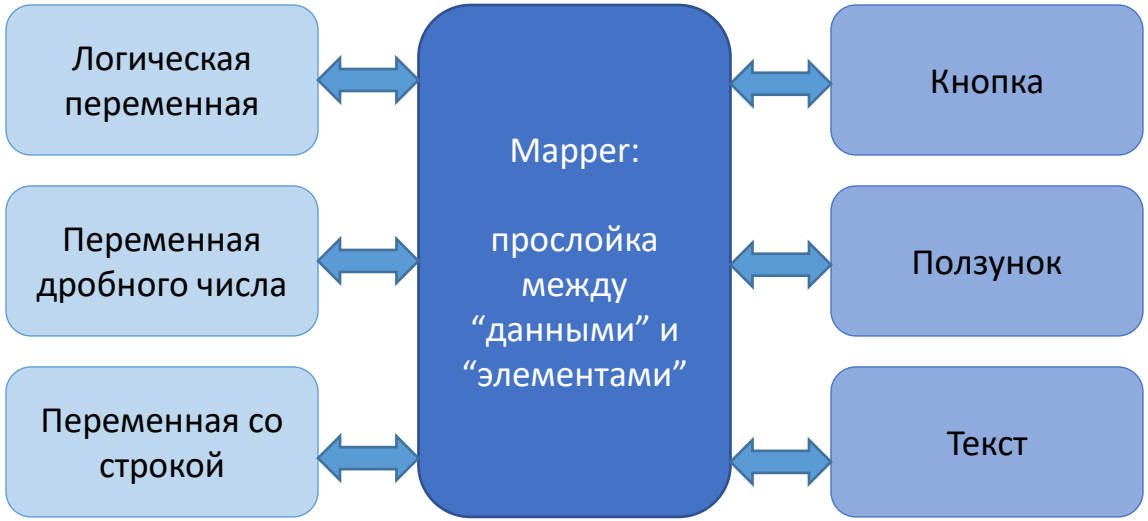
Пример работы анимации



Пример работы сцена

Пользовательский интерфейс





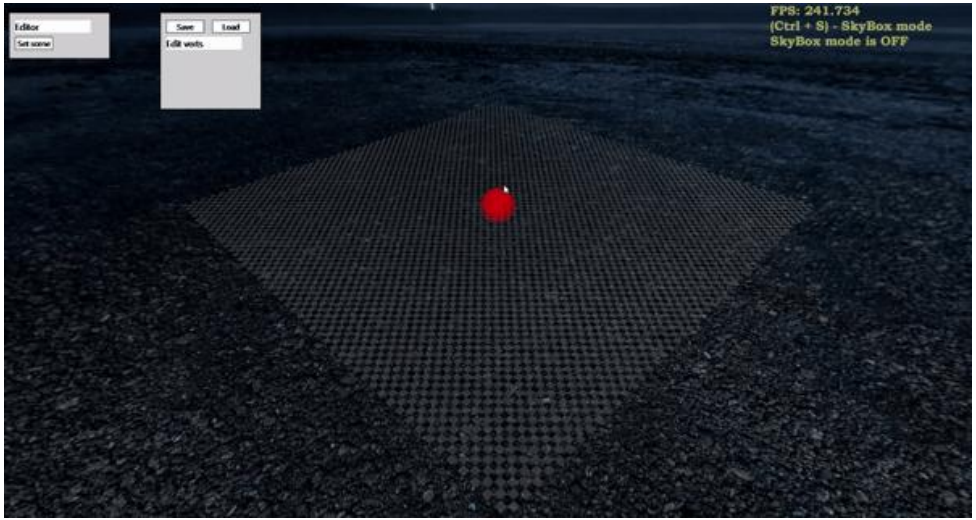
Представление пещер. Редактор графа

Режимы редактора:

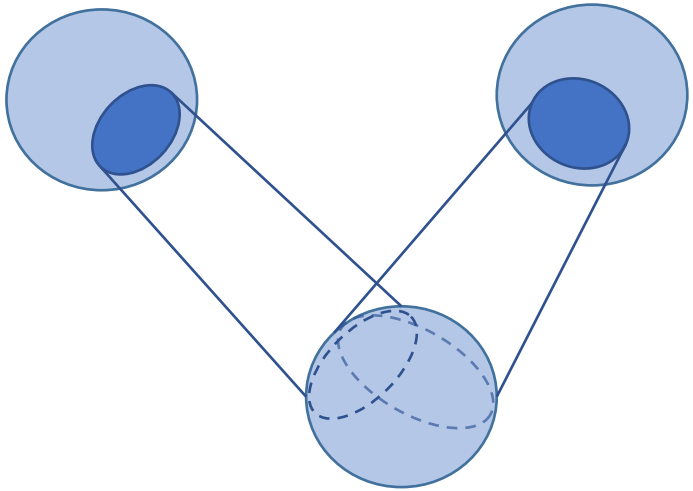
- Редактирование набора вершин графа
- Редактирование набора ребер графа
- Редактирование позиций вершин графа
- Редактирование радиусов вершин графа

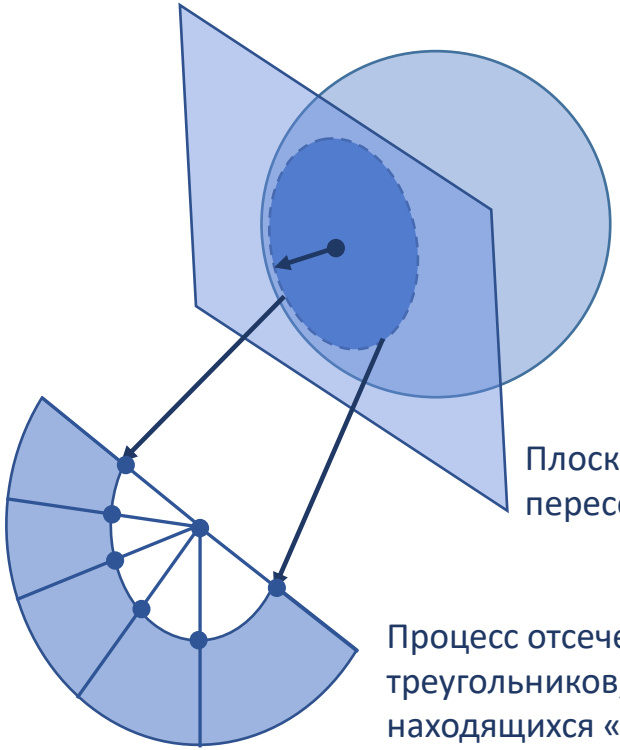
Графы можно сохранять в файлы и загружать вновь для редактирования.

По графу можно построить геометрию пещер.

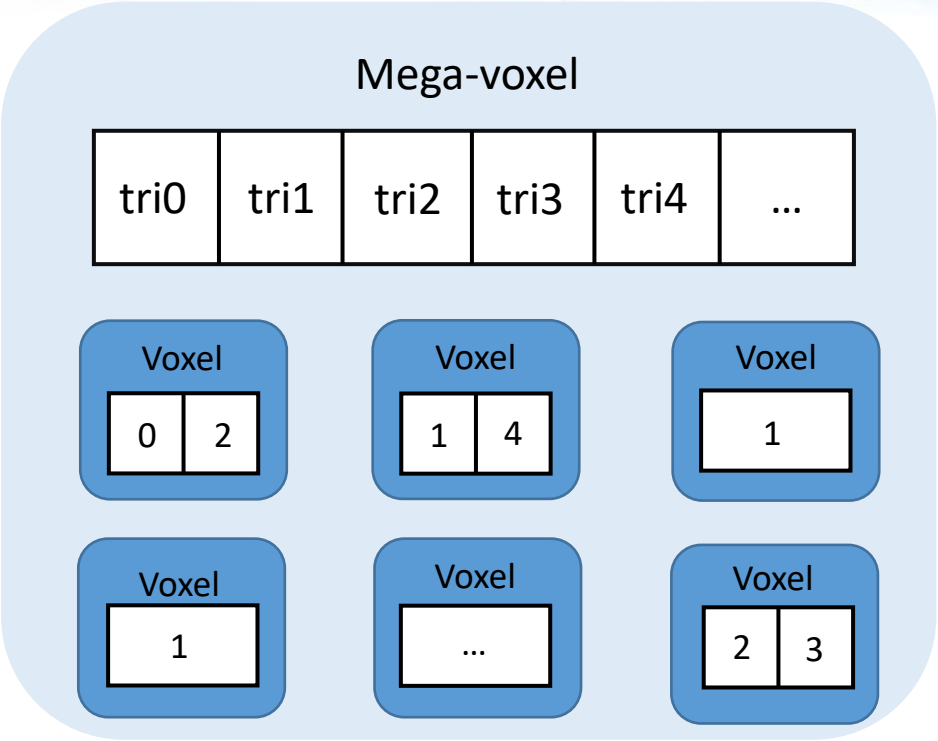


Пещеры. Формирование геометрии

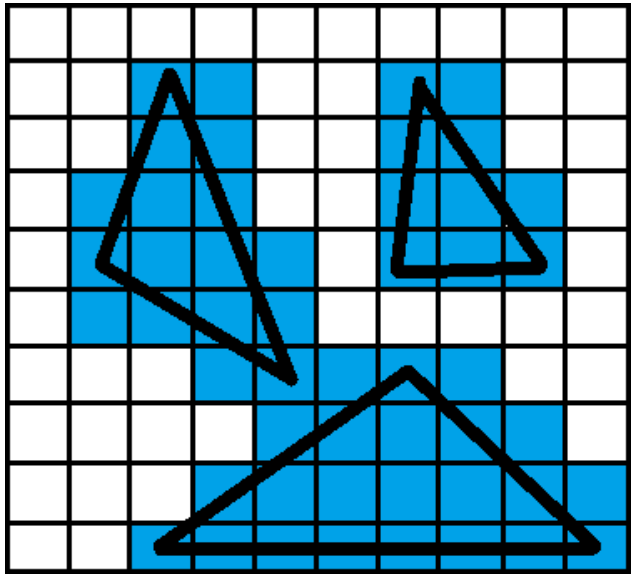




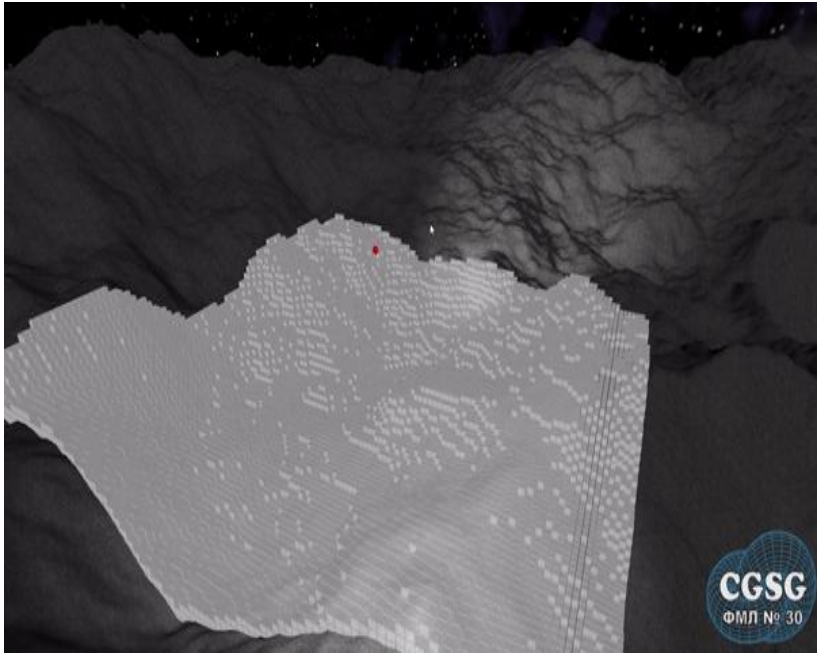
Воксели. Вокселизация пространства.



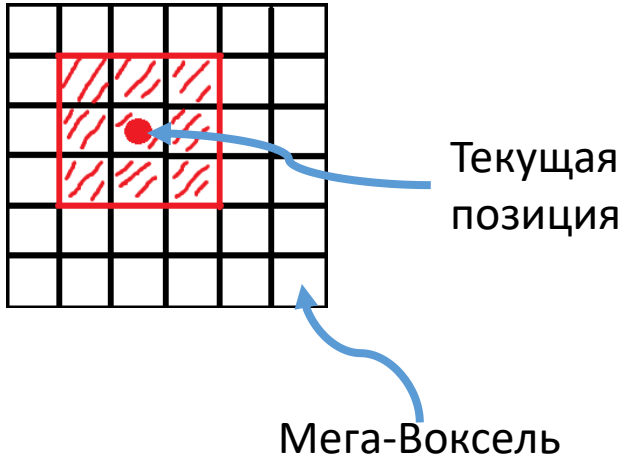
Стадия накопления



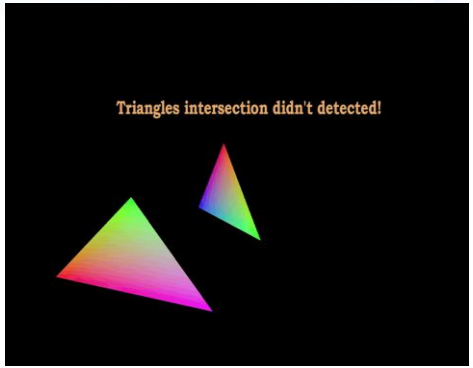
Вокселизация пространства.

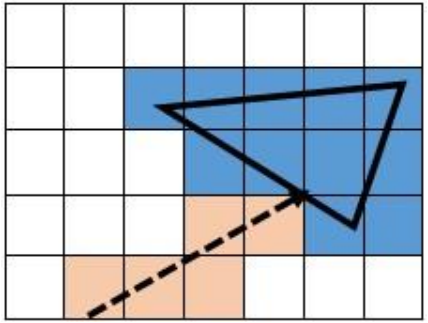


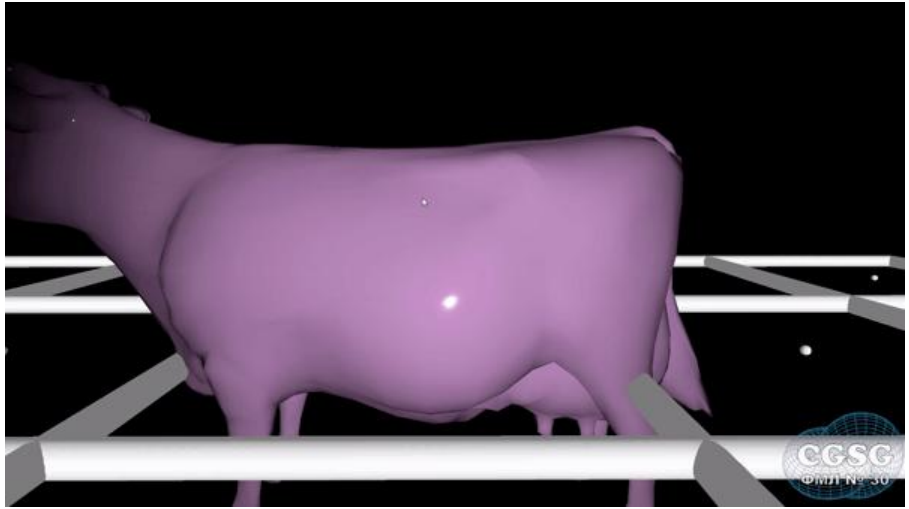
Определение используемых мега вокселей



Сцена. Статические коллизии







Заключение

- Цели и задачи, поставленные в ходе реализации проекта, выполнены успешно.
- Результат проекта – приложение, моделирующее и визуализирующее пещеры.
- Успешно реализована идея создания крупного проекта большой группой авторов.
- Мы уверены, что наш опыт реализации подобного проекта будет полезен в будущих исследованиях моделирования, а также поможет улучшить визуализацию сложных редактируемых объектов в реальном времени.

Разработка кода

- **Худяков Тимофей 09-4 класс**
работа над ядром вулкана, системой маркеров, статических коллизий
- **Ильясов Александр 10-3 класс**
работа над системой анимации, сцен, моделей, вокселизацией пространства
- **Витман Илья 11-2 класс**
работа над системой материалов, освещения, визуализация теней
- **Пашков Иван 11-5 класс**
работа над ядром вулкана, системой текстур, примитивов, пользовательского интерфейса, редактором графа
- **Топунов Антон 11-6 класс**
работа над ядром вулкана, системой конвейера вывода изображения
- **Фисунова Лидия 11-6 класс**
работа над системой генерации пещер.

Разработка кода

Авторов: 6

Языки: C++ / Assembly x86-64 / GLSL

(вершинные, тесселяционные, геометрические, пиксельные и вычислительные шейдера)

Время разработки: зима-весна 2024-2025

Размер кода: 1.99 MiB

(100 x *.cpp – 889 KiB, 83 x *.h – 726 KiB, 187 x *.glsl – 370 KiB, 1 x *.asm – 3.6 KiB)

Строк кода: 62745

(*.cpp – 27800, *.h – 22800, *.glsl – 12000, *.asm – 145)