



NVIDIA®

**Шумы в компьютерной графике
реального времени**

**Андрей Татаринов,
NVIDIA Developer Technology Team**

План рассказа



- Введение
- Аналитические шумы
- Применение шумов
 - Модификация геометрических моделей
 - Создание объемных эффектов
 - Анимация систем частиц

Введение

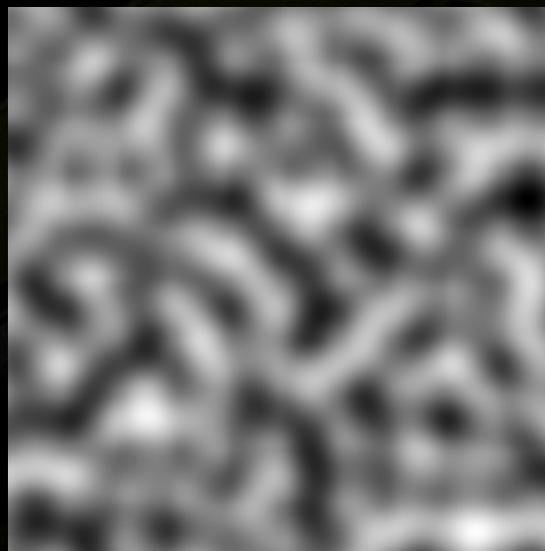


- В настоящее время GPU становятся быстрее и мощнее
- Можно заставить работать в реальном времени алгоритмы, ранее недоступные
- Расчет **аналитических шумов** – один из таких алгоритмов

Аналитические шумы

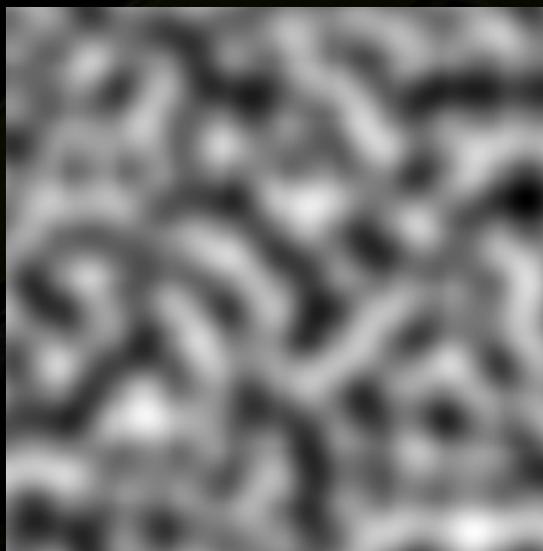


- Псевдослучайная функция, зависящая от всех своих параметров
 - Шумы по значениям
 - Градиентные шумы
 - Шумы со свертками
 - Пространственно-сверточные шумы
 - Комбинированные подходы



Аналитические шумы

- Псевдослучайная функция, зависящая от всех своих параметров
 - Шумы по значениям
 - **Градиентные шумы**
 - Шумы со свертками
 - Пространственно-сверточные шумы
 - Комбинированные подходы

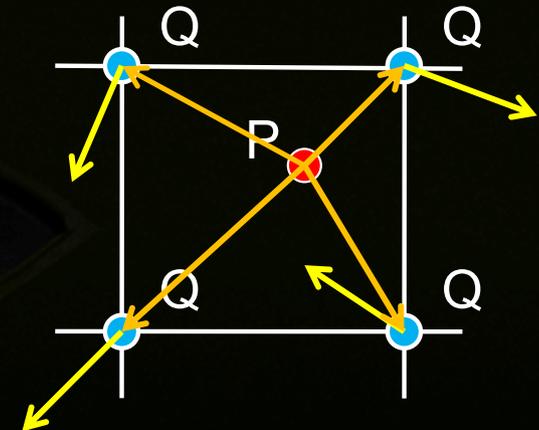
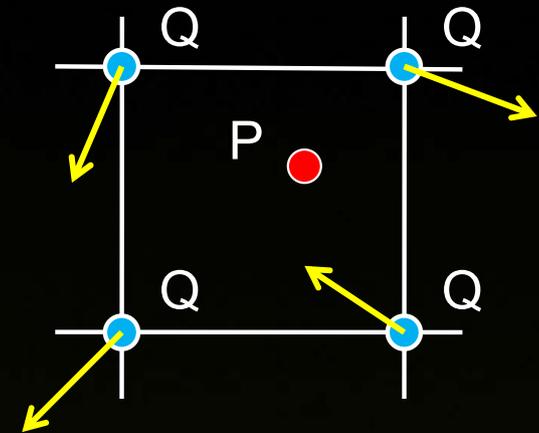


Perlin Noise



Для каждой точки P в пространстве

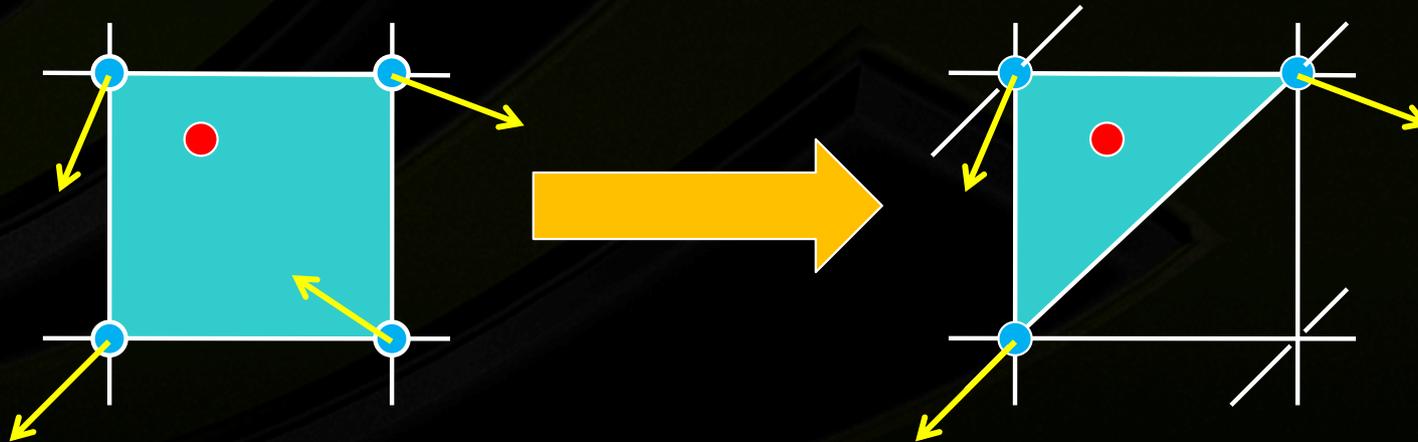
- Взять все ближайшие вершины выбранной регулярной сетки
- Каждой вершине поставить в соответствие псевдослучайный вектор
- Расчитать для каждой вершины скалярное произведение вида $(G, P - Q)$
- Шумовая функция – это взвешенная сумма таких скалярных произведений



Simplex Noise



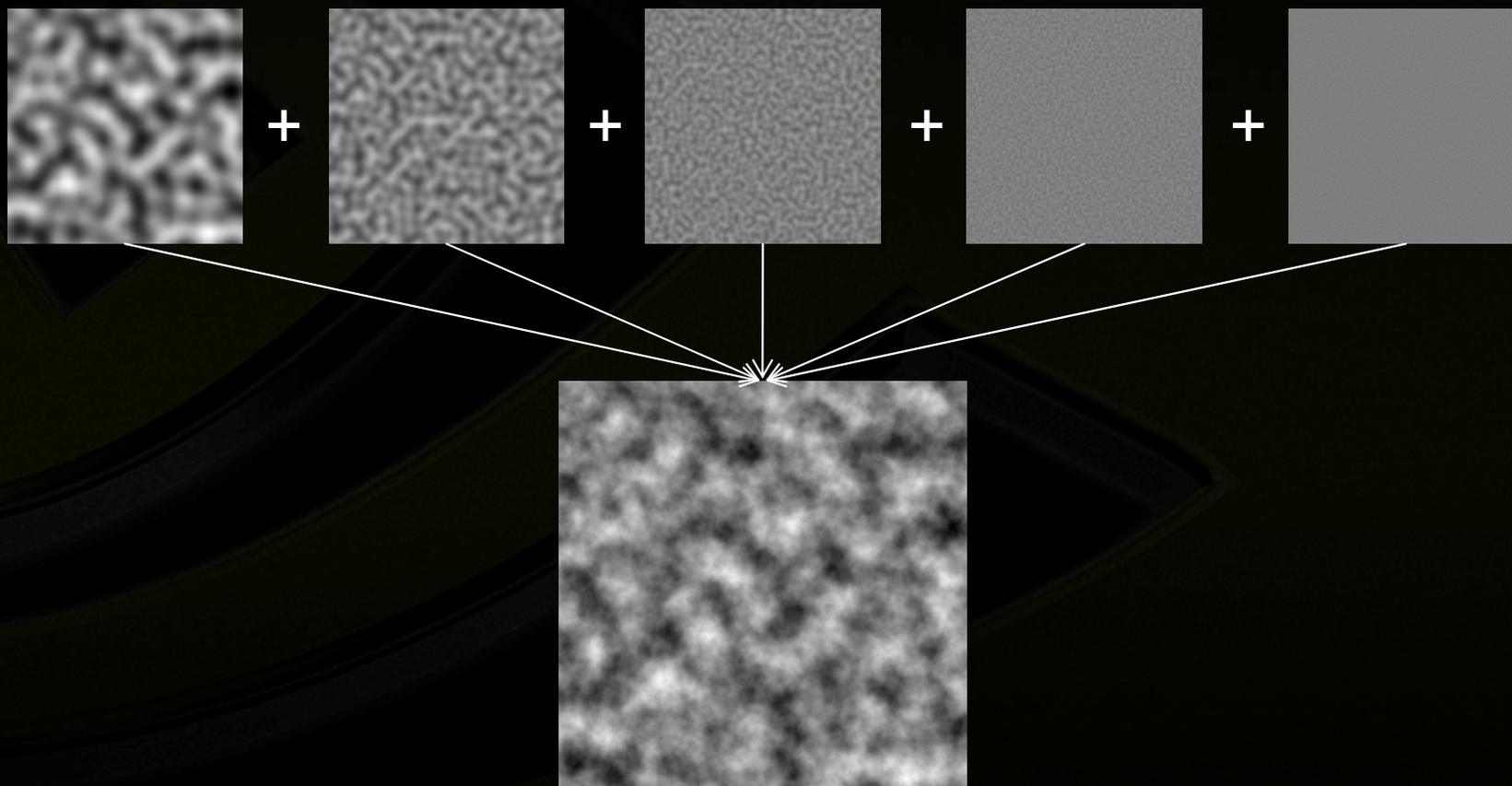
- **Идея – использовать сетку из симплексов**
 - В двумерном случае – треугольник
 - В трехмерном случае – тетраэдр
- **Значительно ускоряет вычисления в случае многомерных шумов**



Процедурные текстуры



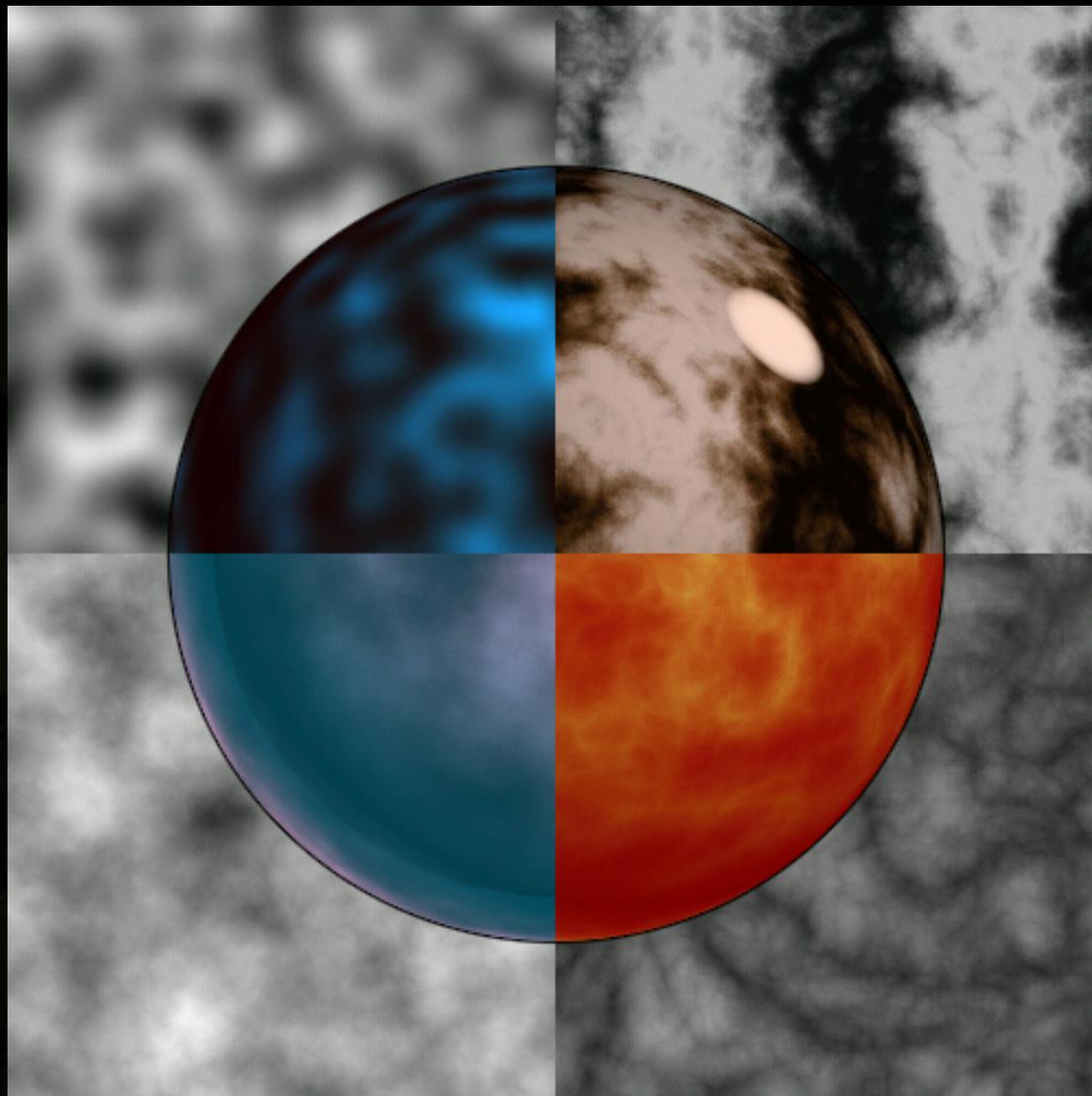
- **Функция турбулентности – сумма нескольких частот шумовых функций**



Процедурные текстуры



noise



$\text{Sin}(x + 1 / f(\text{noise}))$

$1 / f(\text{noise})$

$1 / f(|\text{noise}|)$

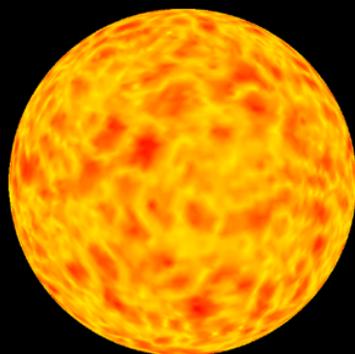
Применение процедурных текстур



■ Дерево



■ Мрамор

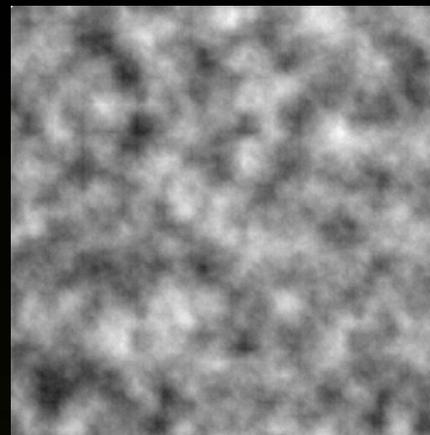


■ Лава

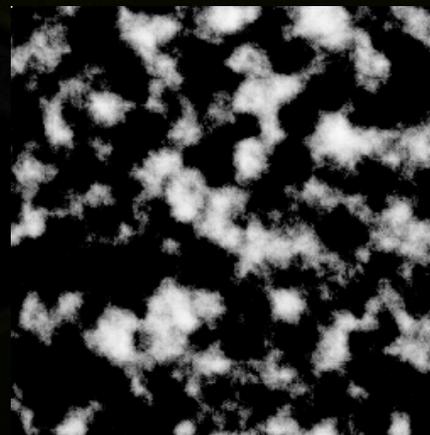
Применение процедурных текстур



Визуализация облаков



$$f' = |f + c|$$



Применение шума



- **Существующие сейчас варианты применения шумов становятся неактуальными**
- **Можно использовать шумы для создания эффектов нового поколения**

Применение шума

- Формирование геометрии
 - Микрорельеф ландшафта
 - Поверхность воды
- Объемные эффекты
 - Огонь
 - Дым
 - Облака
- Системы частиц

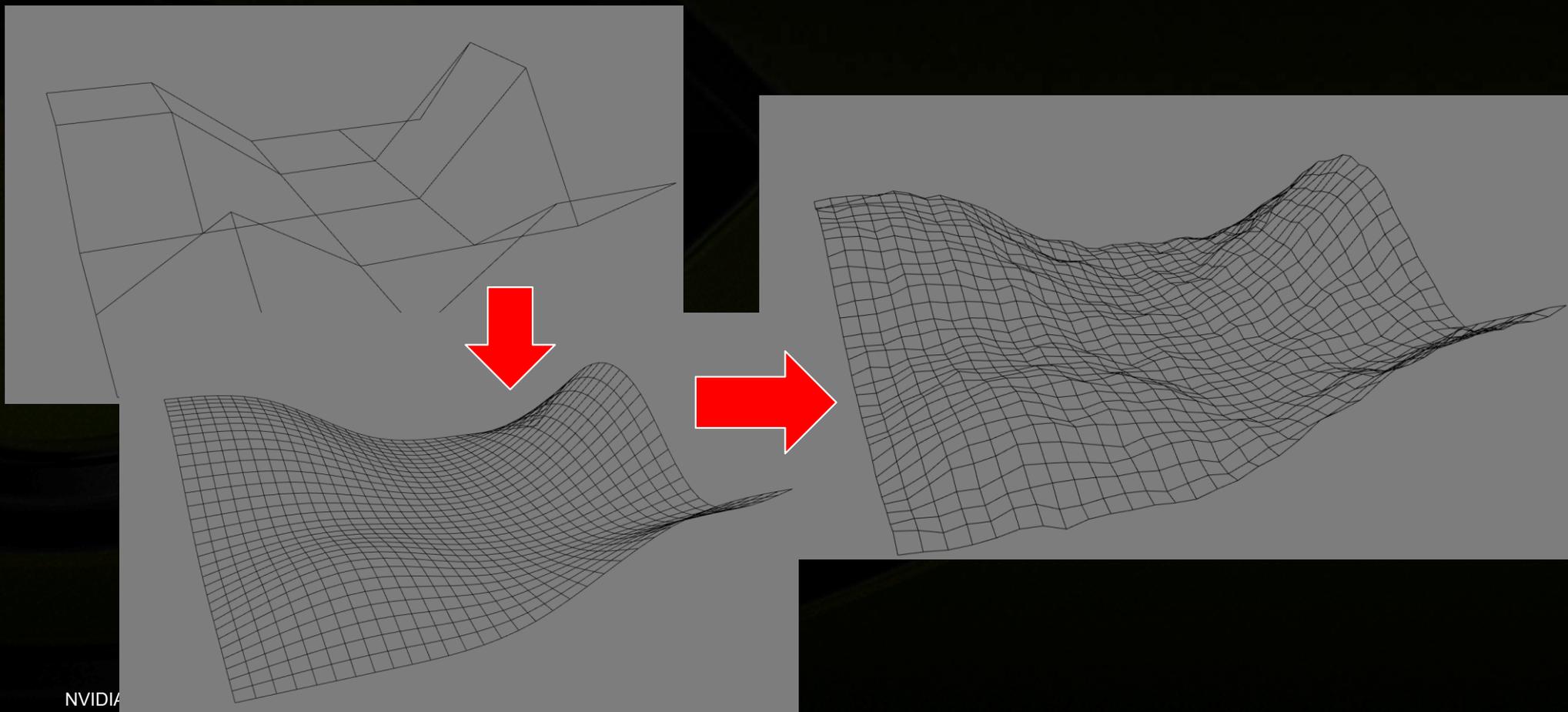
Применение шума

- Формирование геометрии
 - **Микрорельеф ландшафта**
 - Поверхность воды
- Объемные эффекты
 - Огонь
 - Дым
 - Облака
- Системы частиц

Детализация рельефа



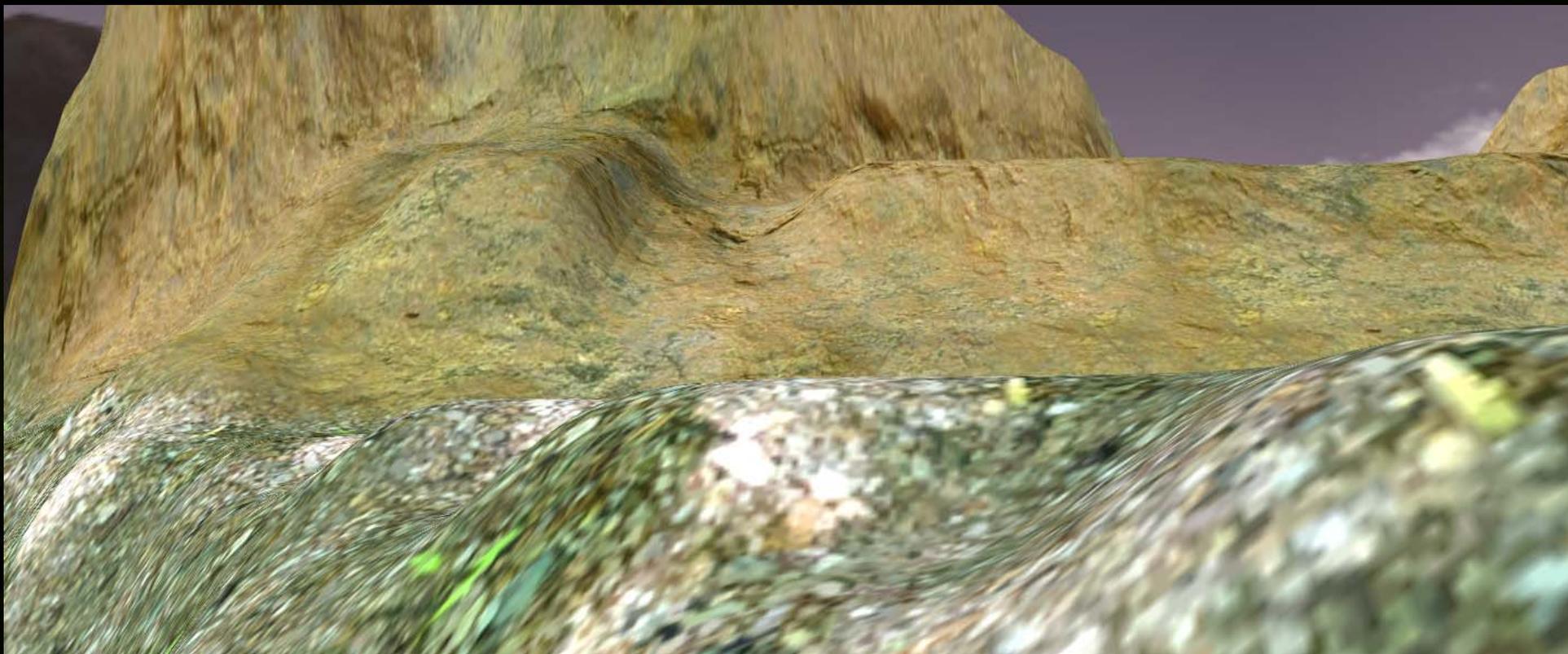
- Использовать тесселяцию и шум для повышения в реальном времени детализации рельефа



Детализация рельефа



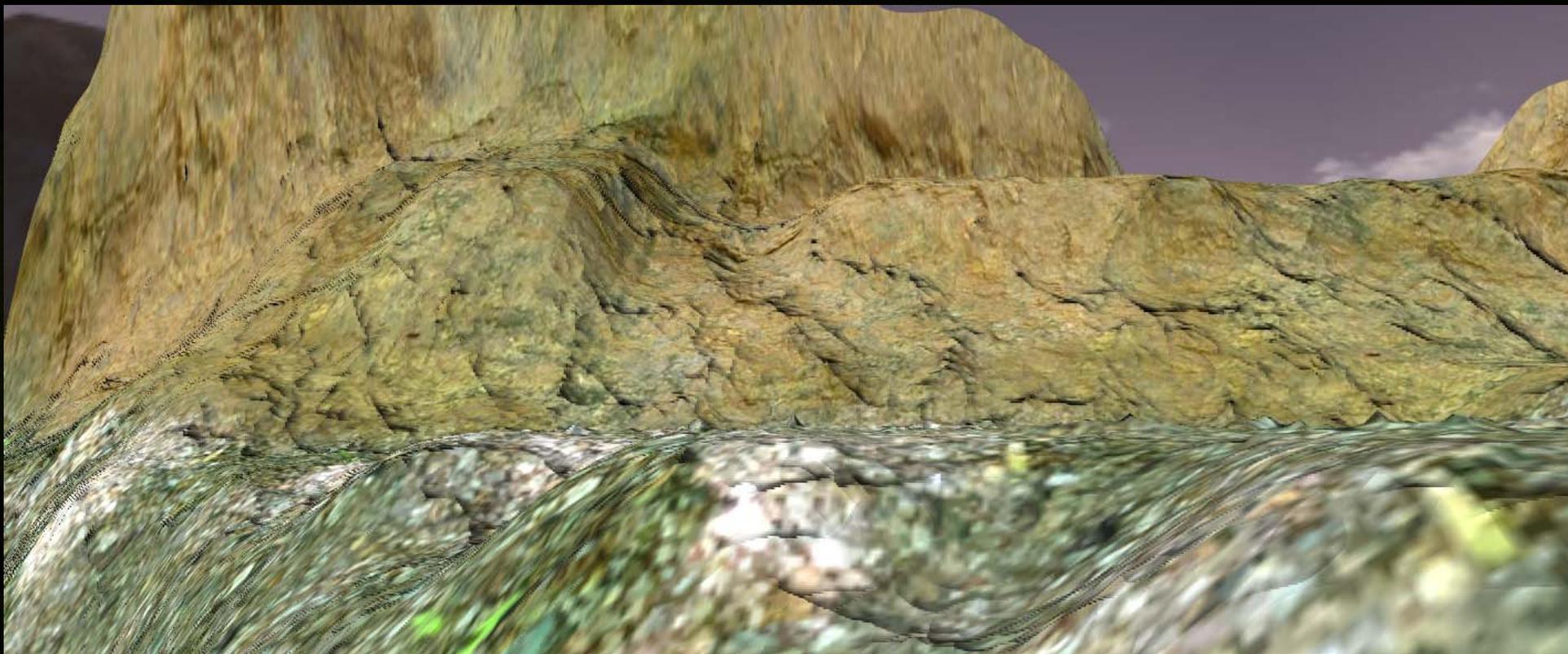
- Повышение детализации
- Экономия памяти



Детализация рельефа



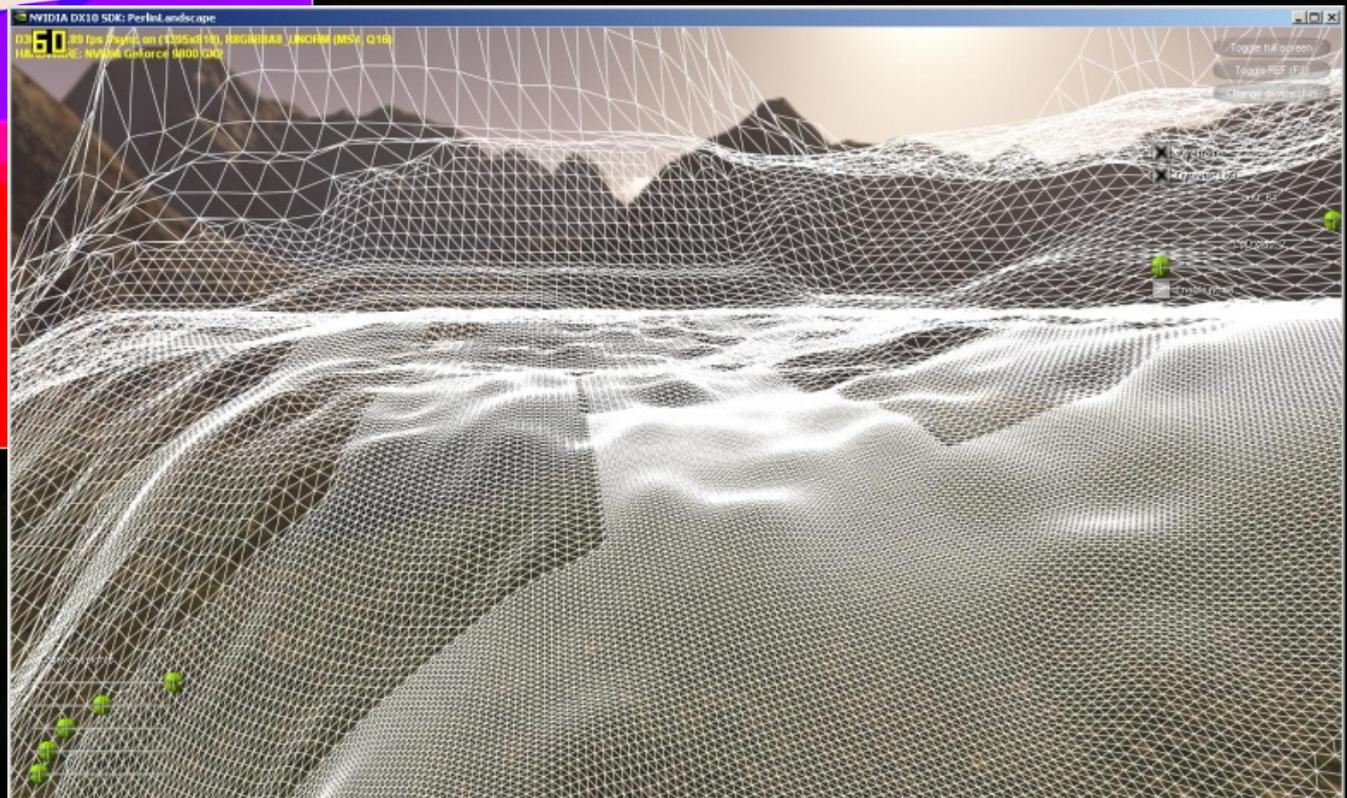
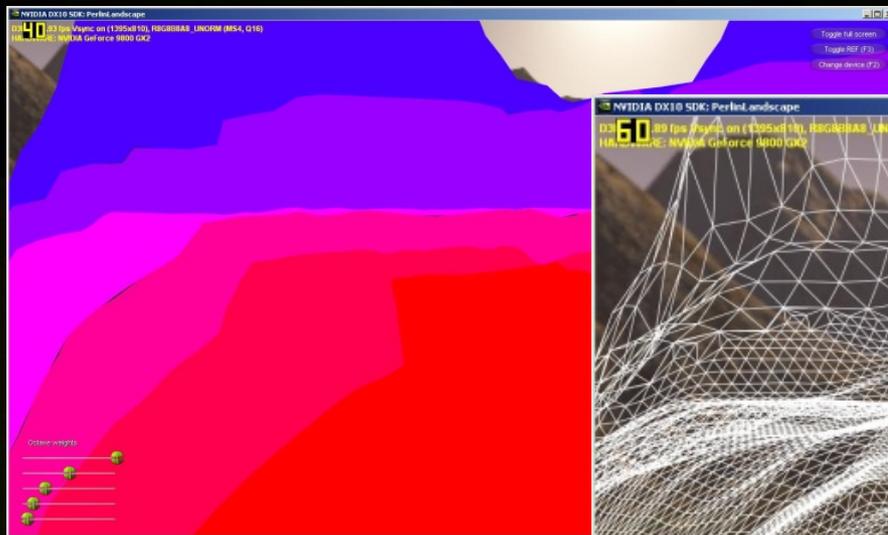
- Повышение детализации
- Экономия памяти



Детализация рельефа



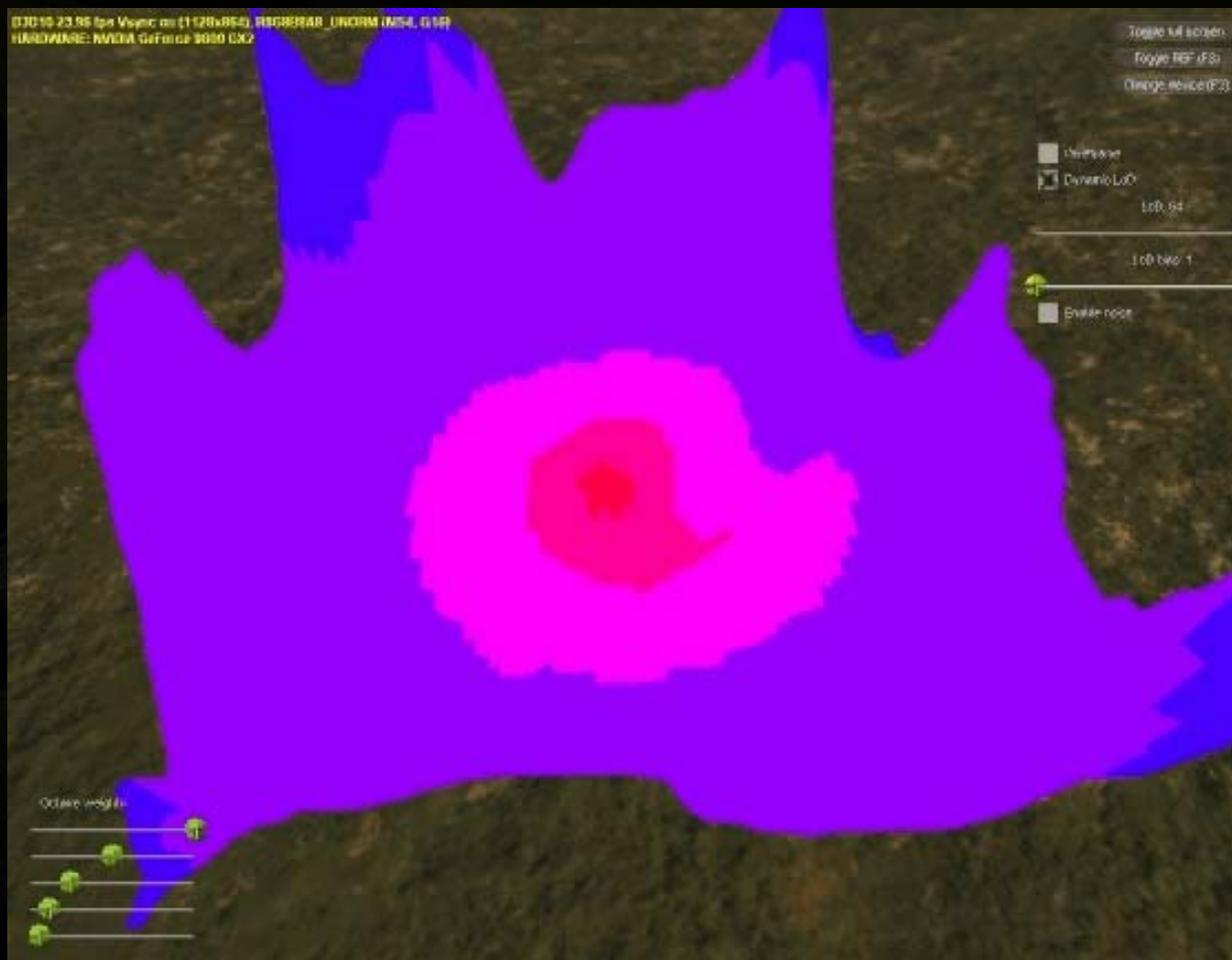
- Ландшафт тесселируется сильнее вблизи наблюдателя



Детализация рельефа



- Ландшафт тесселируется сильнее вблизи наблюдателя



Пример работы



Детализация рельефа

Преимущества:

- Не нужно хранить большую текстуру для всего ландшафта
- Можно изменять уровни детализации динамически

Недостатки:

- Большая вычислительная сложность

Применение шума

- Формирование геометрии
 - Микрорельеф ландшафта
 - Поверхность воды
- Объемные эффекты
 - **Огонь**
 - Дым
 - Облака
- Системы частиц

Традиционный подход

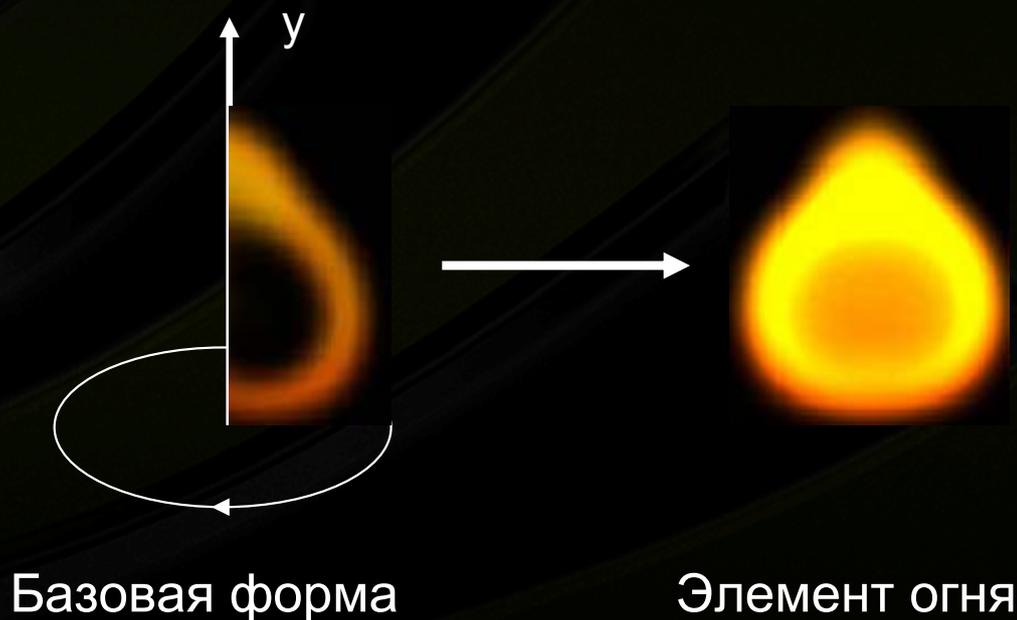


- Построен на базе набора слоев
- Каждый слой содержит видеотекстуру
- При пересечении с объектами возникают артефакты



Объемный огонь

- Взять изображение огня и построить по нему фигуру вращения
- От базового изображения огня зависит внешний вид результирующего эффекта



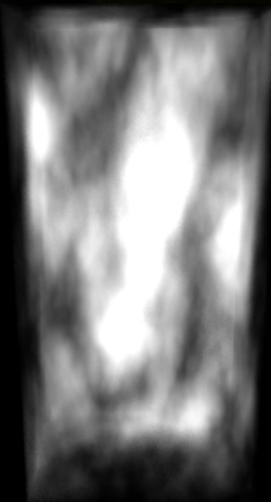
Объемный огонь

- «Зашумить» полученный объем с помощью четырехмерной функции шума



Элемент огня

+



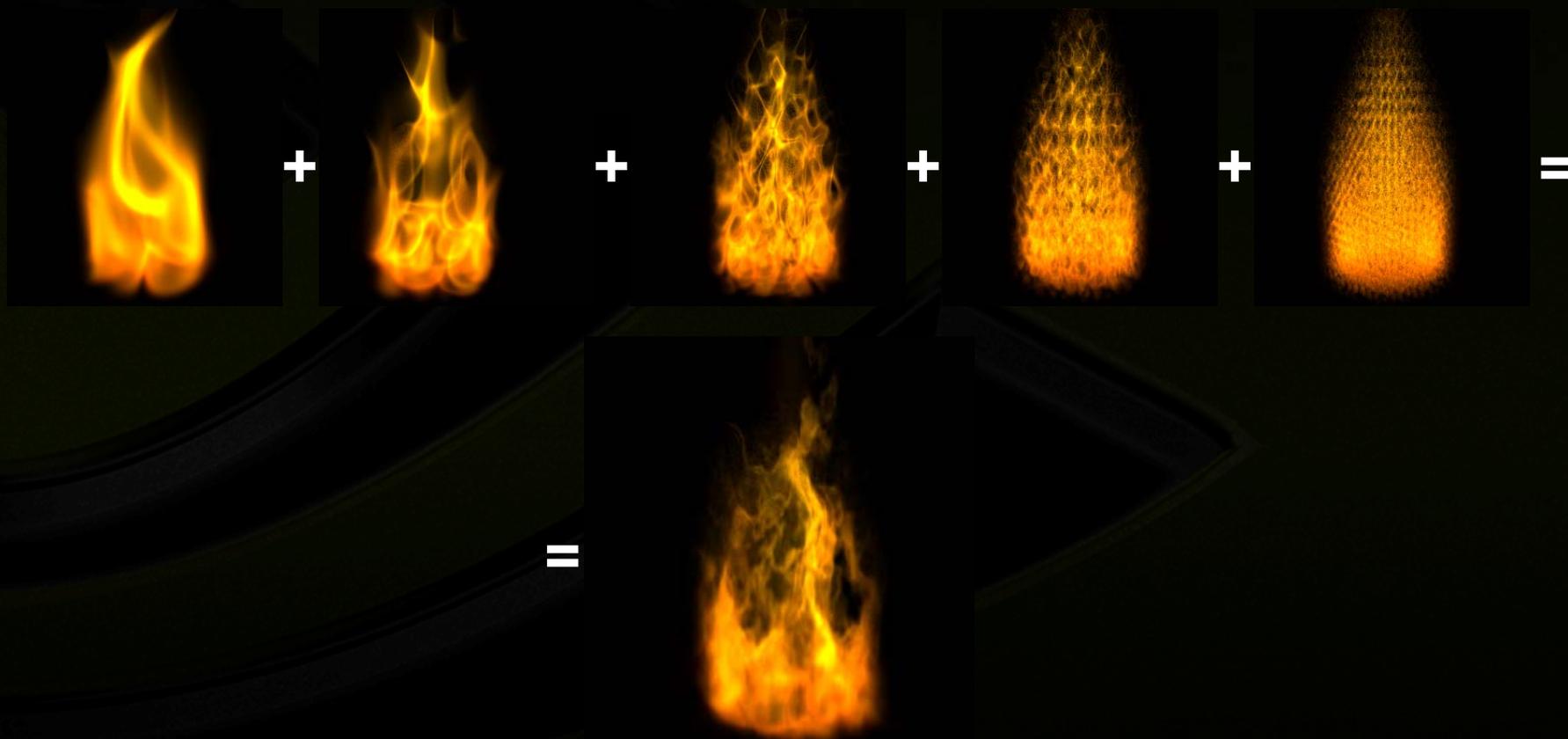
=



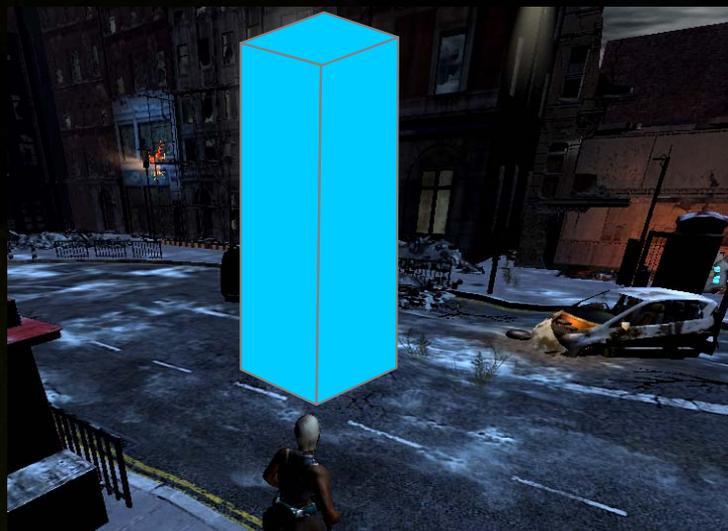
Пламя

Объемный огонь

- Турбулентное поле – взвешенная сумма нескольких частот шума



Визуализация

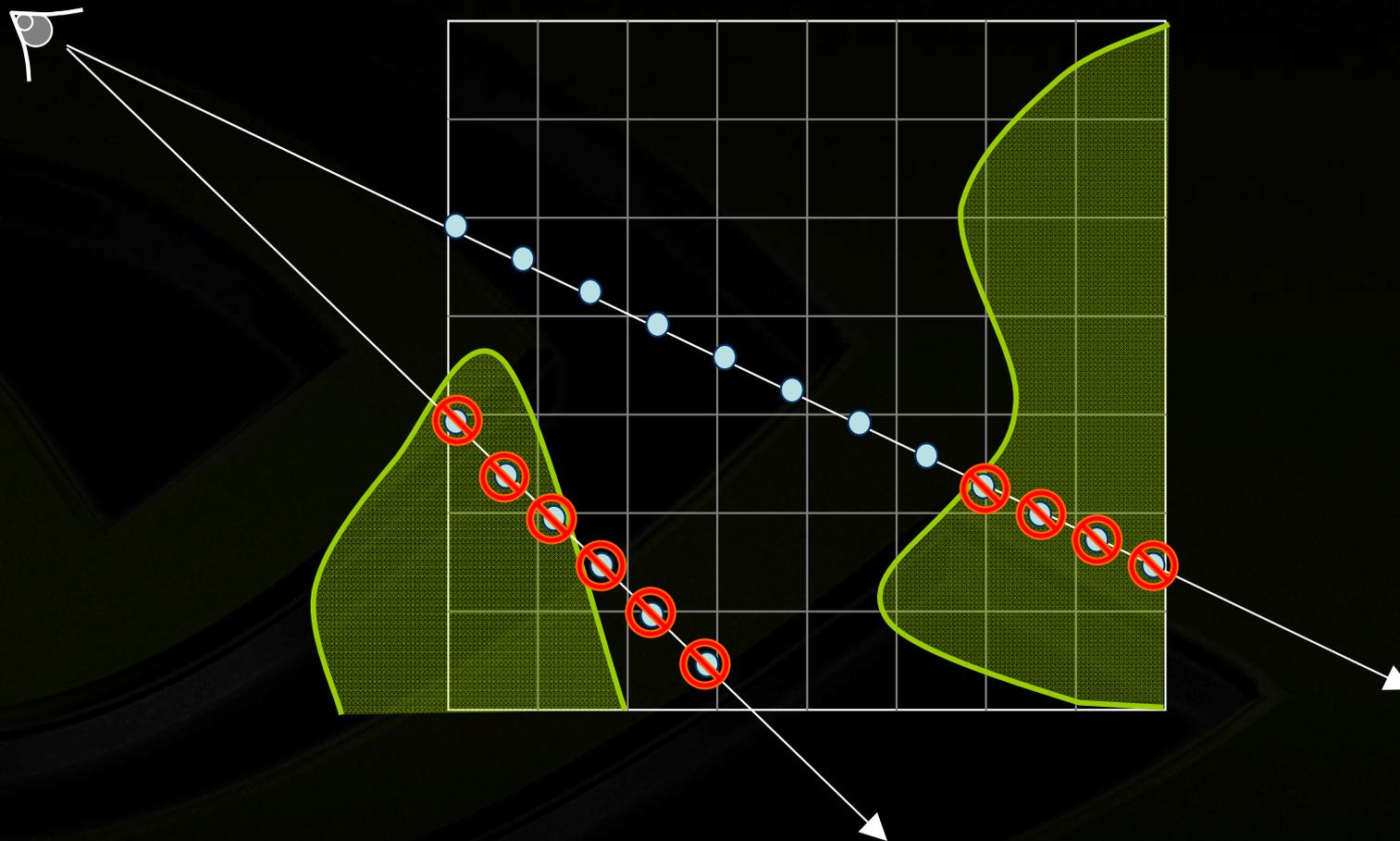


Нарисовать передние грани объема



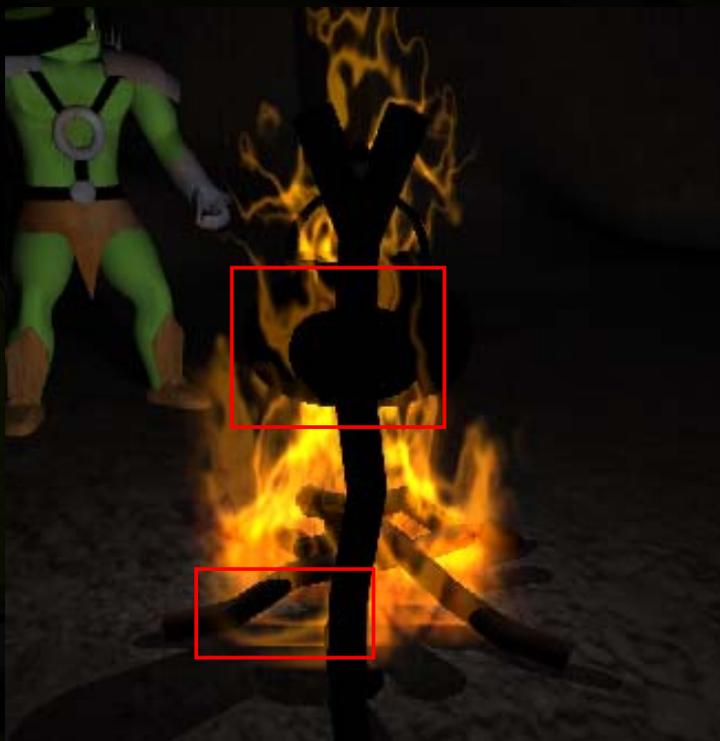
Встроить в сцену

Визуализация



Подавление артефактов

- Использовать случайное смещение начальных позиций лучей для подавления артефактов слоистости



Без смещения



Со смещением

Подавление артефактов

- Для уменьшения «шумности» изображения можно повысить частоту сэмплирования (уменьшить шаг трассировки)



Низкая частота
сэмплирования



Высокая частота
сэмплирования

Пример работы



Объемный огонь



Преимущества:

- Полностью объемный эффект
- Является реалистичной моделью огня
- Не возникает артефактов при пересечении с другими объектами

Недостатки:

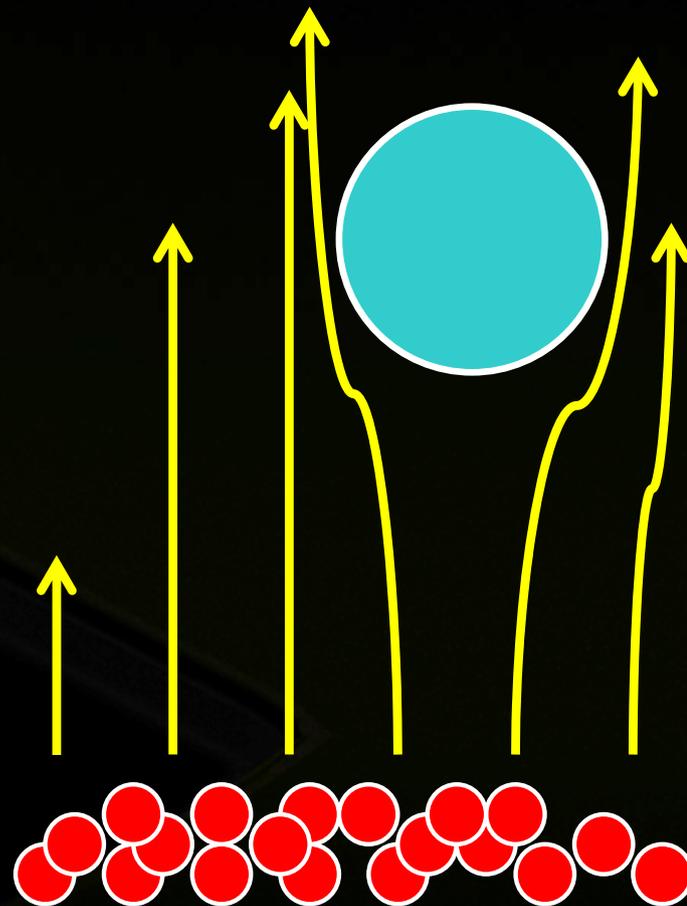
- Не является точной физической симуляцией

Применение шума

- Формирование геометрии
 - Микрорельеф ландшафта
 - Поверхность воды
- Объемные эффекты
 - Огонь
 - Дым
 - Облака
- **Системы частиц**

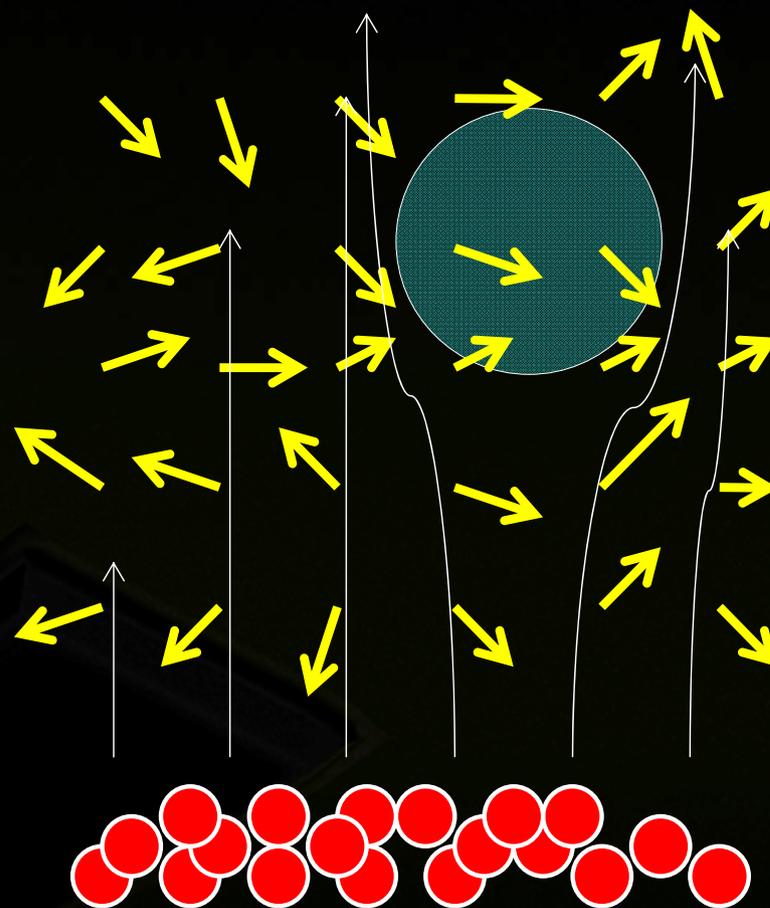
Системы частиц

- Для физически точного моделирования огня можно использовать системы частиц
- Расчет базовой траектории происходит с помощью физической симуляции



Системы частиц

- Чтобы придать частицам характерный турбулентный характер движения, используется шум



Системы частиц



- Визуализация системы частиц напрямую дает нереалистичный результат

D3D10 59.89 fps Vsync on (1024x768), R8G8B8A8_UNORM (MS4, Q16)
HARDWARE: NVIDIA GeForce 9800 GX2

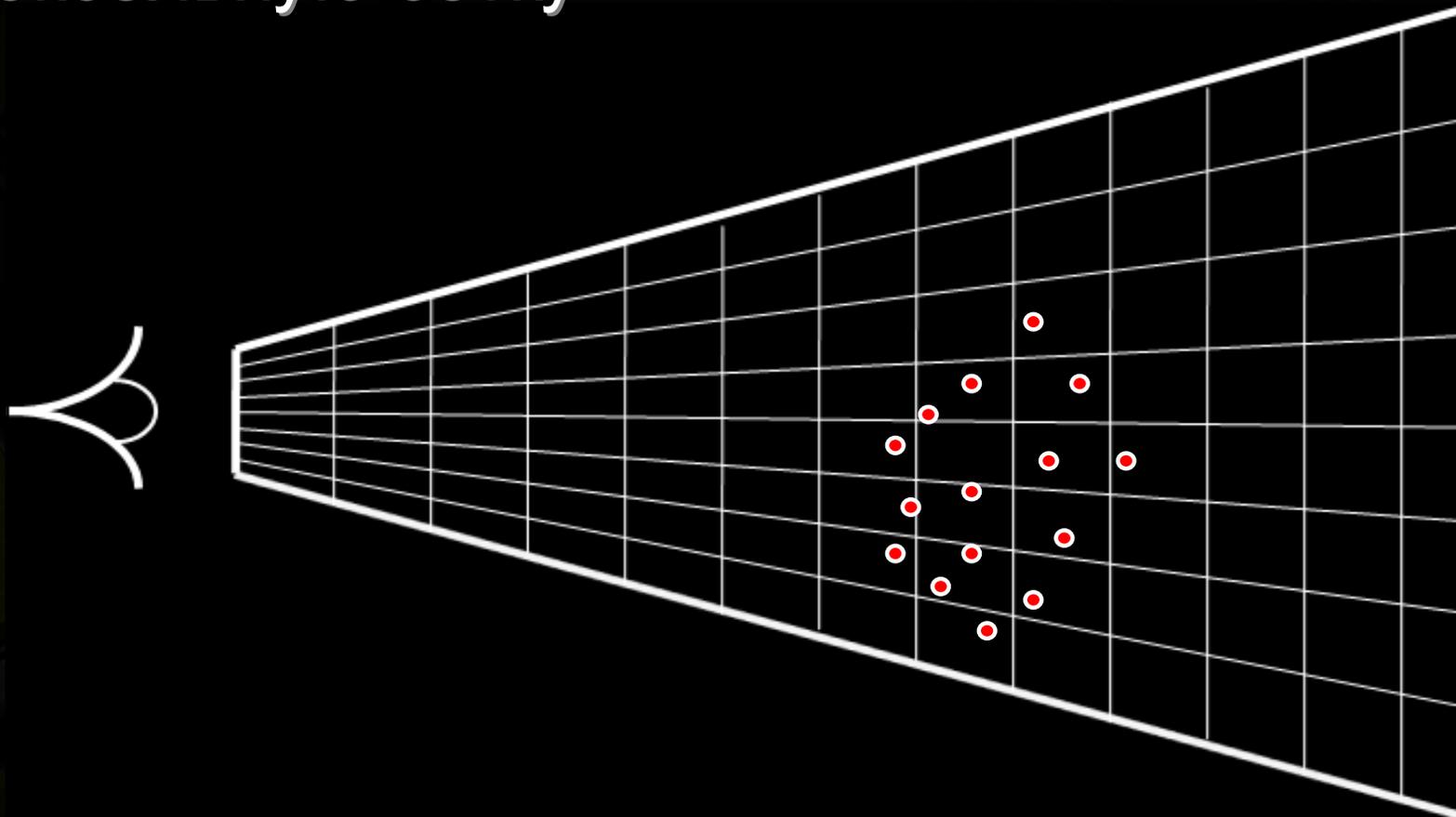


Number of particles: 95000



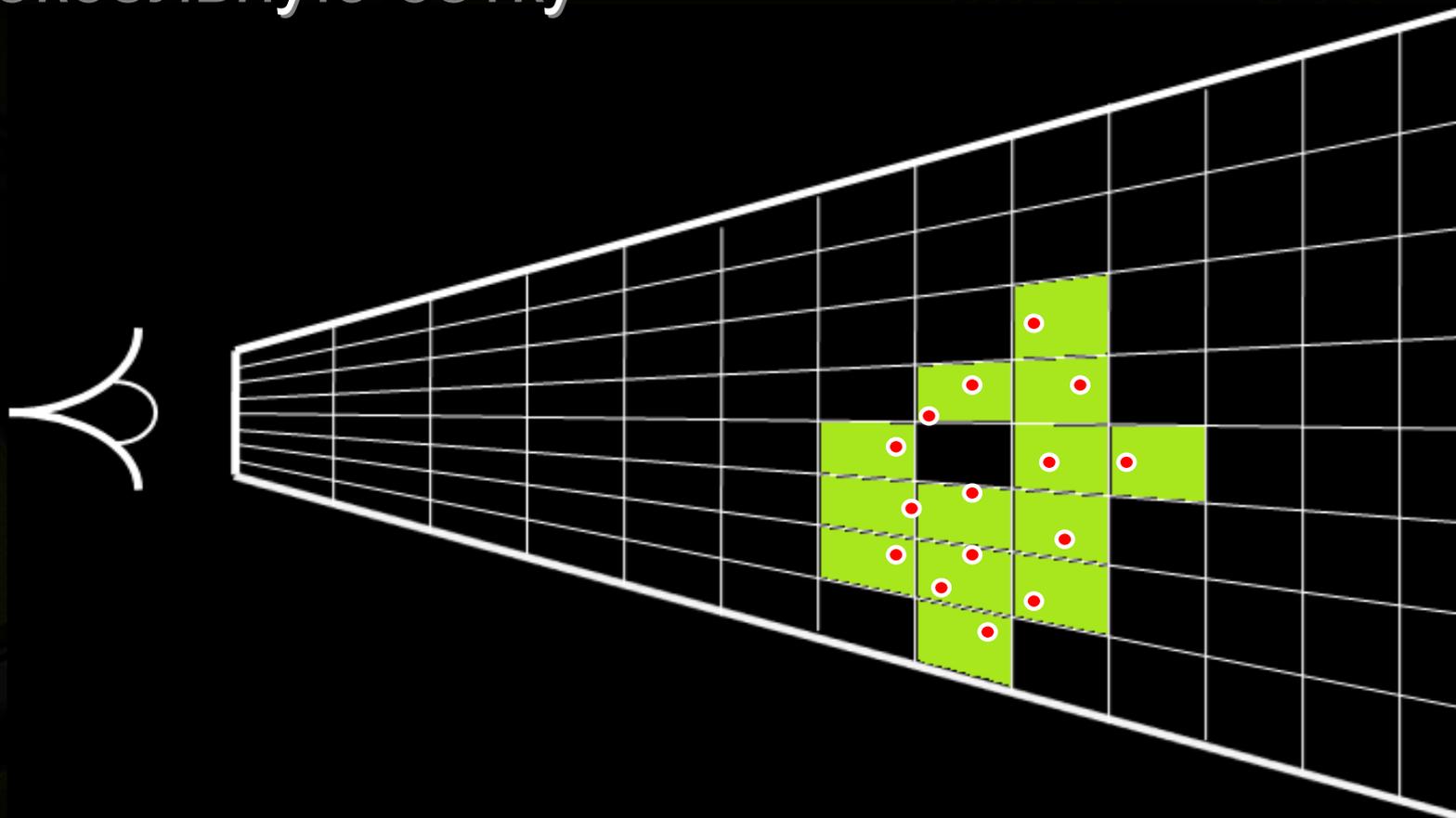
Вокселизация

- Можно спроецировать все частицы на воксельную сетку



Вокселизация

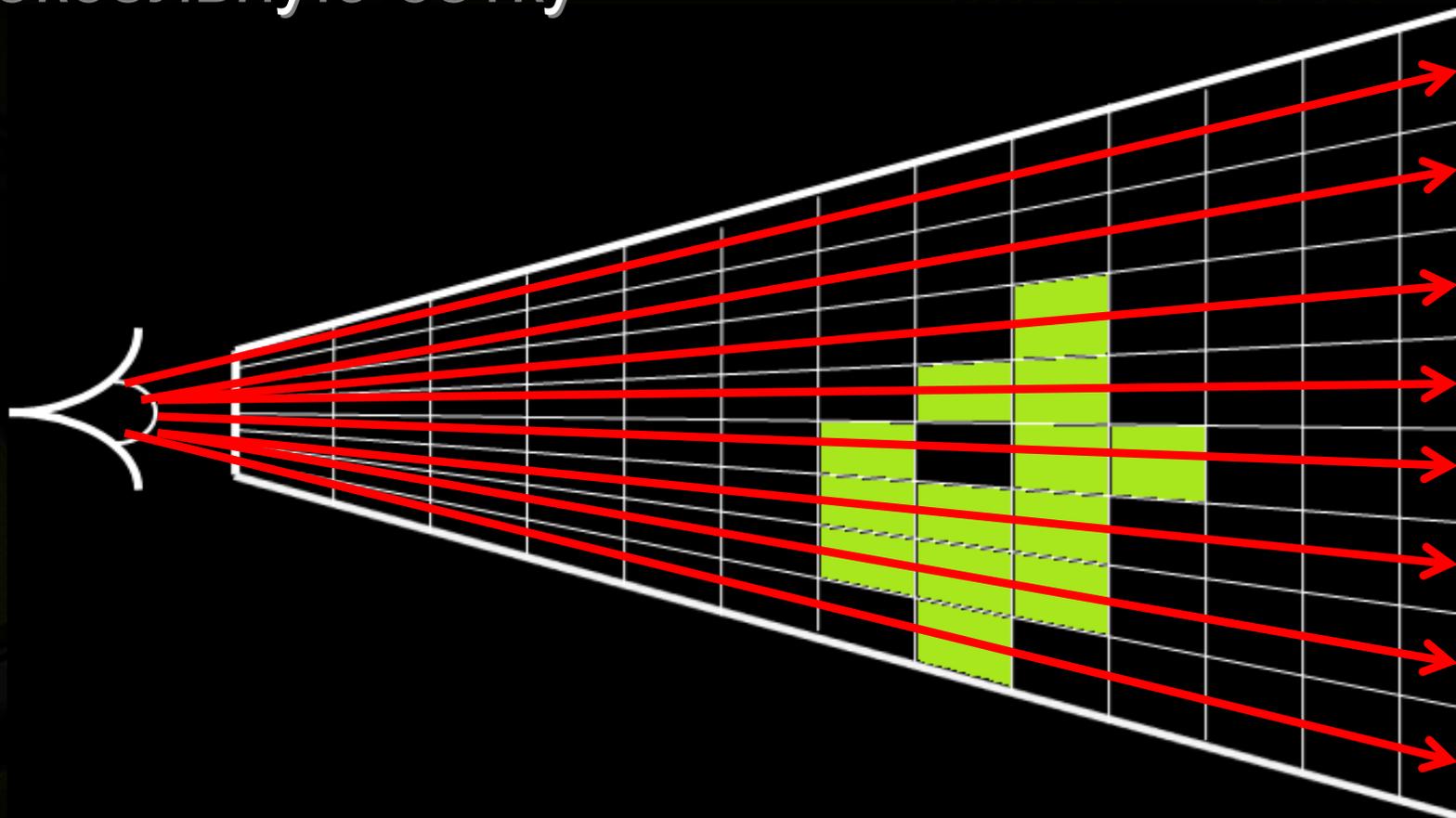
- Можно спроецировать все частицы на воксельную сетку



Вокселизация



- Можно спроецировать все частицы на воксельную сетку



Пример работы



D3D10 28.86 fps Vsync on (1024x768), R8G8B8A8_UNORM (MS4, Q16)
HARDWARE: NVIDIA GeForce 9800 GX2

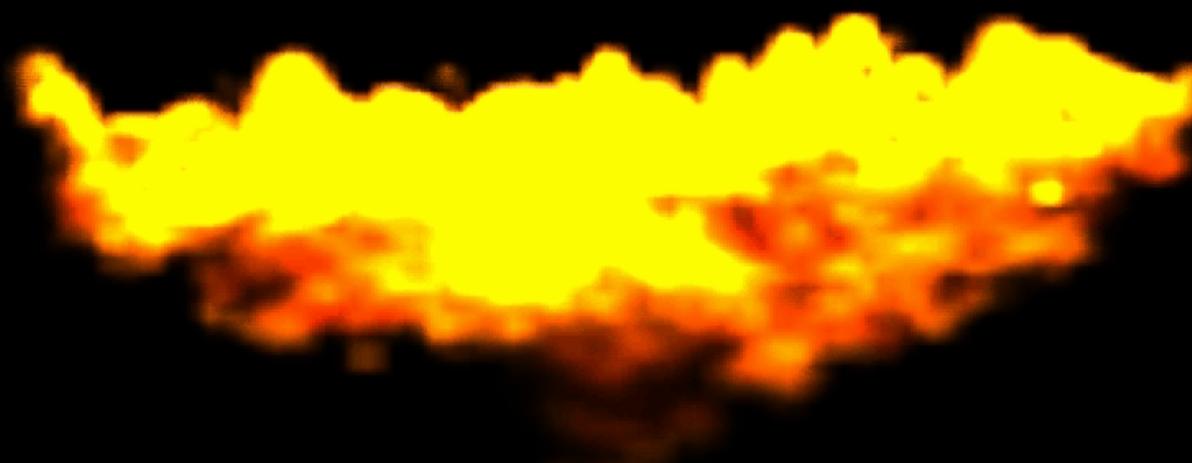
Toggle full screen

Toggle REF (F3)

Change device (F2)

Reset

Number of particles: 160000



Системы частиц



Преимущества:

- Наиболее физически точное и реалистичное представление огня
- Работает быстрее объемного эффекта

Напоследок



- Использование шумов позволяет сделать существующие эффекты качественнее и создать новые эффекты
- Современные GPU позволяют реализовать очень сложные эффекты в реальном времени
- **Огромный простор для экспериментов!**

Спасибо!

