

УДК 004.9

## ОПТИМИЗАЦИЯ ВИРТУАЛЬНЫХ СЦЕН НА UNREAL ENGINE С ПРИМЕНЕНИЕМ ТЕХНОЛОГИИ LEVELS OF DETAIL

Войленко Ю. П., магистрант Отделения информационных технологий

Научный руководитель: Коровкин В.А., ст. преподаватель ОИТ

Национальный исследовательский Томский политехнический университет

С развитием современных технологий виртуальные и интерактивные сцены становятся все более востребованными в различных сферах от игровой индустрии до архитектурного проектирования. Создание качественных и производительных виртуальных миров требует не только творческого подхода, но и глубоких знаний в области оптимизации для достижения баланса между высоким качеством визуального представления и эффективным использованием ресурсов *Graphics Processing Unit (GPU)*.

Целью данной статьи является представление методологии оптимизации виртуальных сцен на базе Unreal Engine 5 с применением технологии *Levels of Detail (LOD)*, что позволит снизить потребление ресурсов компьютера без потери качества картинки и количества кадров в секунду (*FPS*).

Прежде всего, необходимо провести анализ потребления ресурсов сцены, на которой будет проводиться оптимизация и тестирование, это поможет выявить узкие места и определить приоритетные направления оптимизации.

На первом этапе работы была собрана часть виртуальной сцены в *Unreal Engine 5* из готовых наборов моделей, проведен анализ потребления ресурсов *GPU* с использованием встроенного профайлера, а также измерен показатель *FPS* и скорость визуализации одного кадра.

Таблица 1. Входные данные сцены.

Кол-во объектов	Кол-во треугольников	FPS в точке 1	FPS в точке 2	FPS в точке 3	Скорость визуализации в точке 1	Скорость визуализации в точке 2	Скорость визуализации в точке 3
1246	11 млн	53.59	79.58	73.38	18.66	12.56	10.24



Рис. 1. *ProfileGPU* - потребление ресурсов *GPU* без применения *LOD*

*ProfileGPU* - это инструмент профилирования, который предоставляет детальную информацию о времени, затраченном на визуализацию каждого кадра игры на графическом процессоре (*GPU*). Он позволяет разработчикам видеть, какие части игровой сцены занимают больше всего времени на визуализацию, что позволяет выявить узкие места и проблемы производительности.



Рис. 2. Внешний вид сцены с точками замера *FPS* и скорости визуализации одного кадра.

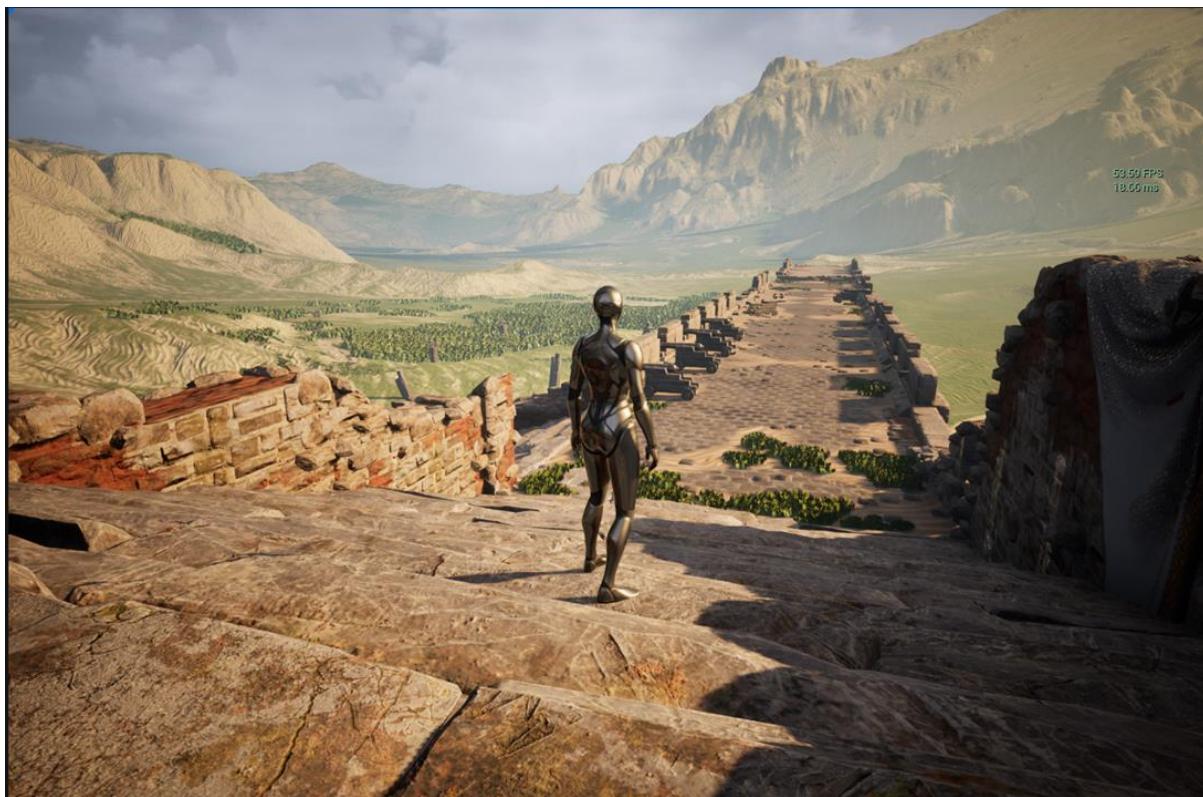


Рис. 3. Показатель *FPS* в точке 1 составил 53.59, скорость визуализации одного кадра составила 18.66 мс. без использования *LOD*.

Следующим шагом было применение технологии *LOD* для части объектов, находящихся на сцене.

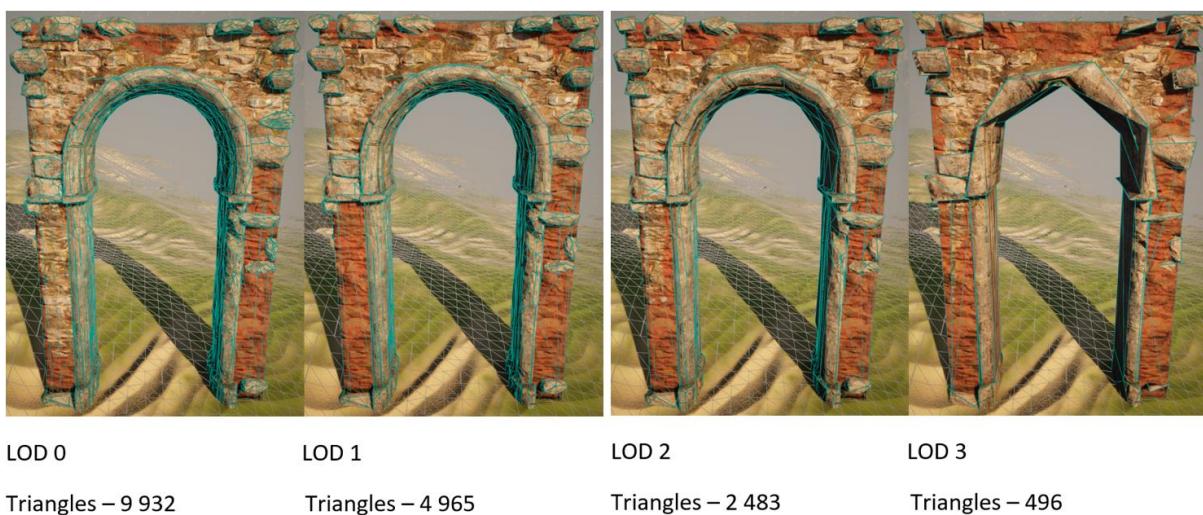


Рис. 4. Пример объекта с использованием технологии *LOD*

Уровни детализации (*LOD*) - это технология, используемая в компьютерной графике, которая позволяет уменьшать количество полигонов моделей в зависимости от их расстояния до камеры. Суть этой техники заключается в том, чтобы использовать более детализированные версии моделей, когда игро-

вой объект находится близко к камере, и переключаться на менее детализированные версии, когда объект находится дальше, тем самым снижая нагрузку на процессор и видеокарту.

В *Unreal Engine 5*, как и в предыдущих версиях, технология *Levels of Detail (LOD)* используется для оптимизации производительности и управления ресурсами при отображении объектов в игровой сцене. Основная идея *LOD* заключается в том, чтобы заменять детализированные версии объектов на более простые и меньше ресурсов затратные версии, когда объект находится далеко от камеры.

В *Unreal Engine 5 LOD* работает по следующему алгоритму:

1. Создание различных версий моделей: Для каждого объекта в игровой сцене разрабатываются несколько версий моделей с разной степенью детализации. Обычно это модели с разным количеством полигонов.

2. Определение расстояний: Для каждого объекта определяется оптимальное расстояние от камеры, на котором должен происходить переход между различными уровнями детализации.

3. Переключение *LOD*: В процессе отрисовки сцены *Unreal Engine 5* автоматически переключает между различными уровнями детализации в зависимости от расстояния от камеры до объекта. Когда объект находится близко к камере, используется более детализированная версия модели, а когда объект находится далеко, используется менее детализированная версия.

Особенности *LOD* в *Unreal Engine 5*:

1. Автоматическое управление: *Unreal Engine 5* автоматически управляет переходом между различными уровнями детализации, что упрощает процесс разработки и оптимизации игровой сцены.

2. Гибкость настройки: Разработчики могут настраивать параметры *LOD* для каждого объекта в сцене, чтобы достичь оптимального баланса между качеством визуализации и производительностью.

3. Визуальное плавное переключение: Переход между различными уровнями детализации происходит плавно и незаметно для игрока, что позволяет сохранить непрерывный визуальный опыт.

После оптимизации выбранных объектов на сцене, был проведен повторный анализ потребления ресурсов *GPU* с использованием встроенного профайльера, а также измерен показатель *FPS* и скорость визуализации одного кадра.

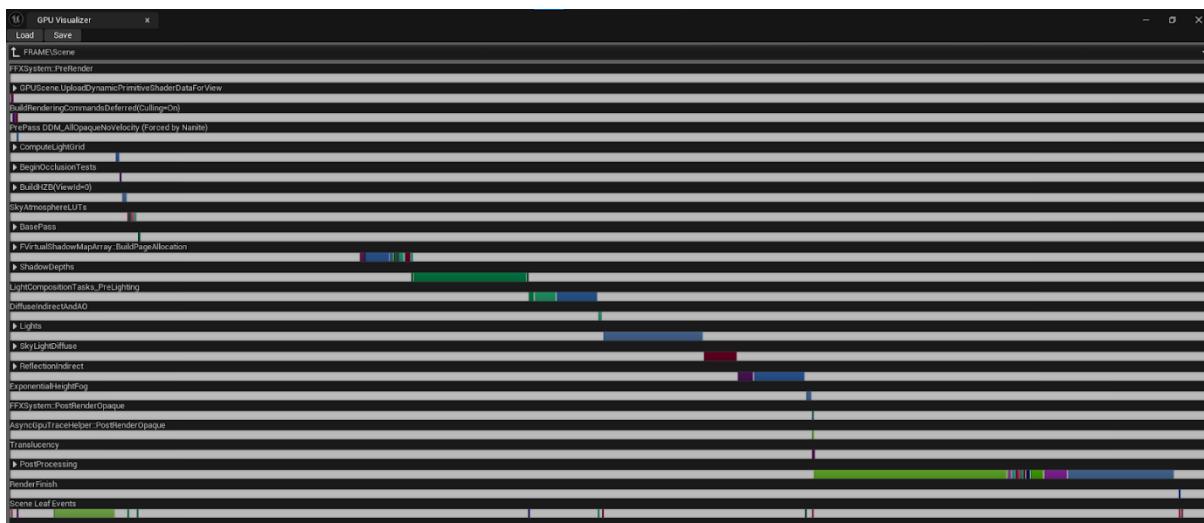


Рис. 5. *ProfileGPU* - потребление ресурсов *GPU* с применением *LOD*.

На рисунке 4 видно, что после применения *LOD* потребление ресурсов *GPU* снизилось в следующих пунктах: *ShadowDepths*, *PostProcessing*.

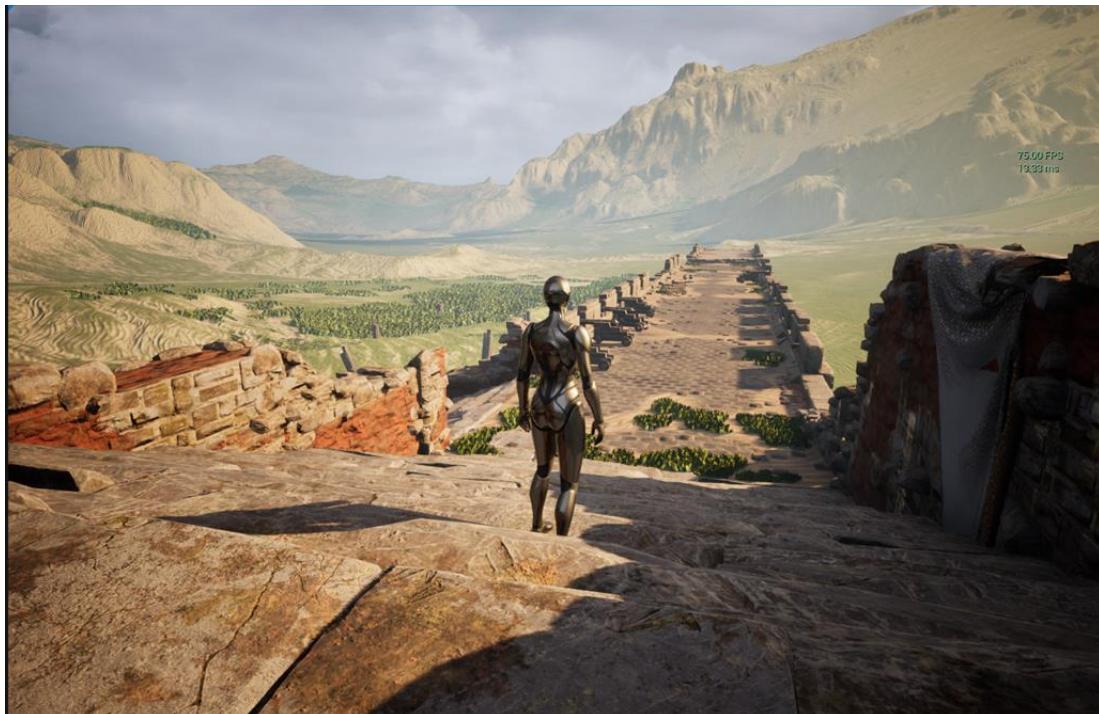


Рис. 6. Показатель *FPS* на точке 1 составил 75.00, скорость визуализации одного кадра составила 13.33 мс. без использования *LOD*.

Таблица 2. Сравнение входных данных и полученных данных сцены.

	FPS в точке 1	FPS в точке 2	FPS в точке 3	Скорость визуализации в точке 1	Скорость визуализации в точке 2	Скорость визуализации в точке 3
входные	53.59	79.58	73.38	18.66	12.56	10.24
полученные	75.00	82.37	76.84	13.33	12.24	9.84

Тестирование проводилось на персональном компьютере со следующими параметрами: *CPU - Intel Core i5-6500, GPU - GeForce NVIDIA 1060 6 Гб, ОЗУ - 16 Гб.*

Исследование проводилось путем анализа потребления ресурсов *GPU* с помощью встроенного профайлера и измерения ключевых метрик производительности, таких как *FPS* и скорость визуализации одного кадра. После применения технологии *LOD* к части объектов на сцене был проведен повторный анализ, который показал значительные улучшения: общая нагрузка на *GPU* снизилась на 1%, *FPS* увеличился на 12.14%, а скорость визуализации одного кадра улучшилась на 11.66%. Важно отметить, что полученные значения являются средними арифметическими по трем точкам.

Таким образом, результаты исследования подтверждают эффективность применения технологии *LOD* для оптимизации виртуальных сцен в *Unreal Engine 5*. Этот подход может быть полезен разработчикам в создании более производительных и качественных виртуальных миров в различных областях, от игровой индустрии до архитектурного проектирования.

### Список литературы

1. Mendez, E., Forsyth, D., & Luebke, D. Level of Detail for 3D Graphics. - Burlington: Morgan Kaufmann Publishers, 2003. - 414 c.
2. Schneider, P. J. GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics. - Addison-Wesley Professional, 2002. - 592 c.
3. Eberly, D. H. 3D Game Engine Design A Practical Approach to Real-Time Computer Graphics. - 2 изд. - Boca Raton: CRC Press, 2007. - 1040 c.
4. Foley, J. D., van Dam, A., Feiner, S. K., & Hughes, J. F. Computer Graphics: Principles and Practice.. - Addison-Wesley Professional., 1990. - 1263 c.
5. Shirley, P., & Marschner, S. R. Fundamentals of Computer Graphics. - Boca Raton: CRC Press, 2009. - 785 c.