

This is a Work in Progress project

Information will be updated as the project develops

Plan

Memory Optimization for High-Density Assets in Unreal Engine

The idea is to create multiple scenes that would have high density assets (high poly, high texture quality, grass animation, cloud system, and potentially animations with a lot of keyframes) and optimize using different techniques, so I will have ground truth (the scene that doesn't have any kind of optimization) then I will have multiple similar scenes that would use one specific technique per scene (for comparison with ground truth) and then I will have a scene that will combine all the techniques together (also for comparison with ground truth).

The comparison will be evaluated by the metrics that will be gathered using the unreal profiler (fps, ms per frame, fps 1% lows, memory used in RAM and VRAM) as well as getting feedback on how the scene looks, as optimizing graphics can deteriorate the visuals. Each person will be given 2 pictures (not a video or the scene itself running as fps can be a factor for their answers) that would be like a split screen and a choice between left and right pictures and the middle option would be that the pictures are the same. After presenting pictures, I will present the scene (or video) where now fps will be a factor of their response, they will have the same options of choice as before.

I could reuse the assets from the [wild west project](#) provided by Epic Education Learning and Training, they provided a [video](#) helping with getting ready for this project.

This pack includes 4K textures, Lumen lighting support and almost 400 meshes. One problem is that the meshes come pre optimized, so I will have to remove the LODs and any other kind of optimization provided with each mesh so I can use it as the ground truth.

Techniques

1. Texture compression

Compress the textures essentially reducing the space allocated for textures in memory.

2. Implement LOD and Imposter techniques

Common optimization method, render lower quality assets at larger distance from camera.

3. Create On Demand asset loading

Load only assets that are in camera's view or close to camera, removes unwanted rendering of assets that are not visible (behind camera or behind other textures).

4. Memory profiling/Optimizing

Use unreal tools to monitor memory usage to identify memory leaks, overuse of high-resolution assets and inefficient loading processes. Create own tool if needed.

5. Data Oriented Design

Optimize how data is accessed and processed, attempting to achieve the most out of modern CPU architectures and cache systems.