

DEMO REEL BREAKDOWN

By Aman Sachan

→ METEOROS - Vulkan Cloudscape Rendering (00:04): (C++, Vulkan, GLSL)

- **Realistic real-time** rendering of clouds in under **3ms/frame** on a notebook 1070
- **Responsibilities:**
 - ◆ **Vulkan framework:**
 - 2D and 3D Texture Support;
 - Image Loading and Transition utility;
 - Buffer Management utility;
 - Handling Descriptor sets, Descriptor layouts, and Descriptor pools;
 - Handling Command Generation and passing these commands to the graphics or compute queue;
 - Ping-ponging command buffers, textures, and frame data;
 - Window functionality with mouse and keyboard controls;
 - ◆ **Ray Marching** of Cloud shapes using the 3D and 2D noise textures passed into the shaders along with the remap function and manipulation of noise fbms;
 - ◆ **Optimizations:**
 - **Reprojection** combined with **rendering every 16th pixel**;
 - **Cheap sampling:** only doing expensive operations using high frequency noise once we have determined we are inside a cloud;
 - **Early termination** if opacity threshold is reached;
 - **TXAA and reducing number of samples:** Using TXAA we can reduce the number of samples of the 3D noise textures needed by a factor of 6;
 - ◆ **Post-Processing:**
 - **Temporal Anti-Aliasing:** This removes a lot of artifacts that occur when we reduce the number of samples in our ray march;
 - God Rays;
 - Tone Mapping: color is in HDR space;

→ MONTE CARLO PATH TRACER (00:39): (C++, OpenGL)

- CPU based Path Tracer with a lot of features including:
 - ◆ **Volumetric Rendering;**
 - ◆ **Multiple Importance Sampling;**
 - ◆ **Fresnel reflectance** models;
 - ◆ **Materials:**
 - **Micro-facet** surfaces;
 - Transmissive Materials;
 - Glass Material;
 - Diffuse Surfaces;
 - Specular Surfaces;
 - ◆ **Realistic Light Sources:**
 - Environment Lights (skyboxes)
 - Point Lights
 - Spot Lights
 - ◆ **Thin Lens Camera**
- **Optimizations:**
 - ◆ **BVH Acceleration** (9800% speed up);
 - ◆ **Multi-Threading;**

→ CLUSTERED DEFERRED AND FORWARD PLUS (01:00): (Javascript, WebGL)

- Implemented Clustered Deferred and Clustered Forward Plus Shading Techniques in WebGL
- Real-time (60+ FPS) rendering of more than 2100 dynamic lights in complex scenes
- Optimizations over regular forward shading:
 - ◆ Clustering: A technique of binning and filtering lights based on their location in 3D space;
 - ◆ Deferred Shading: A technique to only carry out calculations for fragments that are visible (have the least depth);
 - ◆ Compacted g-buffer (total of 8 channels): Storing positions, colors, and normals;
 - ◆ 2 Component Normals;

→ HAND OF GOD (01:47): (Unreal Engine 4)

- Hand of God is an asymmetric co-op game **merging** traditional **non-VR and VR gameplay**
- Implemented (using the Blueprints Visual Scripting system):
 - ◆ **AI**;
 - ◆ Player and Enemy **movements**;
 - ◆ **Controls**;
 - ◆ **Weapons**;
 - ◆ Special weapon projectile attacks;
 - ◆ and helped establish **networked gameplay**;

→ INTERESTING LEVEL GENERATOR (02:12): (Javascript, WebGL, GLSL, threejs)

- A **procedural multi-layer level generator** that generates levels based on a voronoi-like graph after it has been heavily modified by various filters to create interesting level layouts;
- These filters were in-place to remove things like intersections between paths and rearranging paths for interesting level design (inter-looping paths but not too many loops that it's a jumbled mess)
- Also Implemented:
 - ◆ **Realistic Fog** shader;
 - ◆ Biome & Elevation dependent **terrain shader**;
 - ◆ Controllable **Crumbling Pathway aesthetic** via instancing;