Overview
https://zhuanlan.zhihu.com/p/33881505

GPU Driven Pipeline 概要介绍 - 安伯霖
https://zhuanlan.zhihu.com/p/37084925

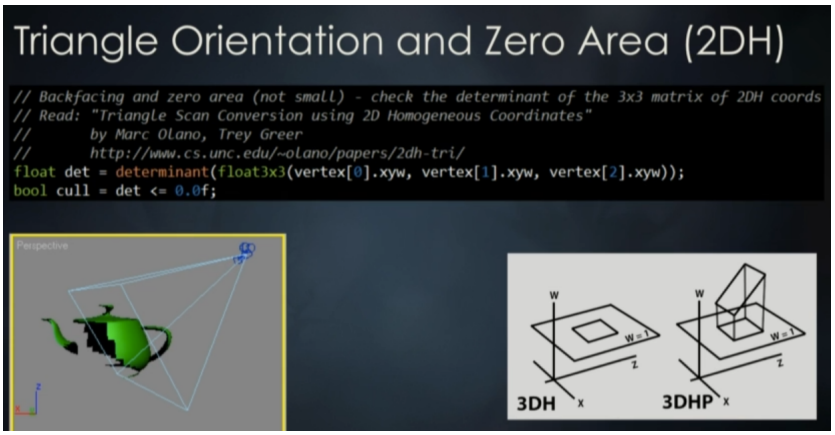
主要参考资料（网盘均有存储）
- siggraph15, < GPU-Driven Rendering Pipelines> @ ubisoft
- gdc16, < Optimizing the Graphics Pipeline With Compute> @ EA
- gdc18, TerrainRenderingFarCry5 @ ubisoft

GPU Driven Pipeline - MaxwellGeng
https://zhuanlan.zhihu.com/p/44411827

GTC GPU-Driven Rendering Nvidia（pdf in 网盘）
http://120.52.51.14/on-demand.gputechconf.com/gtc/2016/presentation/s6138-christoph-kubisch-pierre-boudier-gpu-driven-rendering.pdf

GTC GPU-Driven Large Scene Rendering（pdf in 网盘）
http://59.80.44.98/on-demand.gputechconf.com/gtc/2015/presentation/S5135-Christoph-Kubisch-Pierre-Boudier.pdf

Optimizing the Graphics Pipeline With Compute
Backface cull



GeometryFX

The GeometryFX library provides convenient access to compute-based triangle filtering (CTF), which improves triangle throughput by filtering out triangles that do not contribute to the final image using a compute based pre-process.
https://gpuopen.com/gaming-product/geometryfx/
https://github.com/GPUOpen-Effects/GeometryFX/

GeometryFX 1.2 – Cluster Culling
Today’s update for GeometryFX introduces cluster culling. Previously, GeometryFX worked on a per-triangle level only. With cluster culling, GeometryFX is able to reject large chunks of the geometry – with corresponding performance increases. Cluster culling is not a new idea – last year at SIGGRAPH, Ubisoft presented a GPU based rendering pipeline which incorporated cluster culling as well.
https://gpuopen.com/geometryfx-1-2-cluster-culling/

Indirect Drawing

Indirect drawing enables some scene-traversal and culling to be moved from the CPU to the GPU, which can improve performance. The command buffer can be generated by the CPU or GPU.

https://docs.microsoft.com/en-us/windows/desktop/direct3d12/indirect-drawing

https://docs.microsoft.com/en-us/windows/desktop/direct3d12/indirect-drawing-and-gpu-culling-

Indirect Buffers iOS Metal

Best Practice: Use indirect buffers if your draw or dispatch call arguments are dynamically generated by the GPU.

https://developer.apple.com/library/archive/documentation/3DDrawing/Conceptual/MTLBestPracticesGuide/IndirectBuffers.htm

Modern GPU Driven Rendering (How to draw fast)

https://github.com/ychding11/GraphicsCollection/wiki/Modern-GPU-Driven-Rendering--(How-to-draw-fast)

Occlusion Culling Refs

https://github.com/ellioman/Indirect-Rendering-With-Compute-Shaders

The Unreasonable Effectiveness of Quasirandom Sequences

Kostas Anagnostou - GPU Driven Rendering Experiments

Kostas Anagnostou - Experiments in GPU-based occlusion culling

Kostas Anagnostou - Experiments in GPU-based occlusion culling part 2

Ulrich Haar & Sebastian Aaltonen - GPU-Driven Rendering Pipelines

Sakib Saikia - Going Indirect on UE3

RasterGrid - Hierarchical-Z map based occlusion culling

StackOverflow - Hierachical Z-Buffering for occlusion culling

Daniel E - Hierarchical Z-Buffer Occlusion Culling with multiple samples

bazhenovc - GPU Driven Occlusion Culling in Life is Feudal

NVIDIA - Siggraph 2014 - Scene Rendering Techniques

Github - nvpro-samples/gl_occlusion_culling

GPU Gems 2 - Hardware Occlusion Queries Made Useful

ARM Developer Center - hiz_cull.cs

L. Spiro - Tightly Culling Shadow Casters for Directional Lights (Part 1)

L. Spiro - Tightly Culling Shadow Casters for Directional Lights (Part 2)

nonoptimalrobot - Shadow Volume Culling

Stephen Hill and Daniel Collin - Practical, Dynamic Visibility for Games

zeuxcg.org - View frustum culling optimization - Representation matters

Occlusion Culling with Hierarchical-Z

https://arm-software.github.io/opengl-es-sdk-for-android/occlusion_culling.html

EXPERIMENTS IN GPU-BASED OCCLUSION CULLING

https://interplayoflight.wordpress.com/2017/11/15/experiments-in-gpu-based-occlusion-culling/