
# New GPU Features of NVIDIA's Maxwell Architecture

Holger Gruen – Senior DevTech Engineer

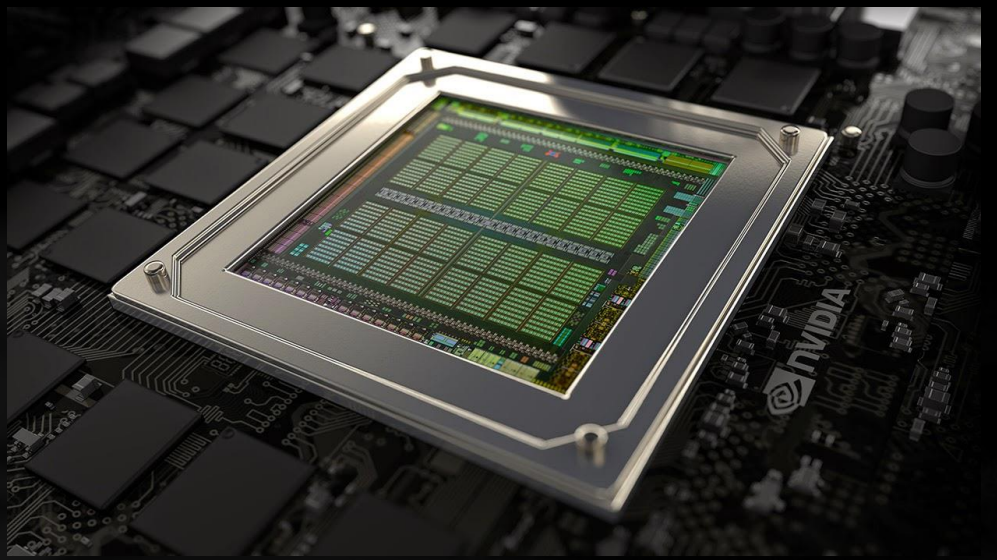
# AGENDA

| | | |
|---|---|---|
| 9:30 am | 10:30 am | Holger Gruen, New GPU Features of NVIDIA's Maxwell Architecture |
| 11:00 am | 12:00 am | Iain Cantlay, NVIDIA SLI and stutter avoidance: a recipe for smooth gaming and perfect scaling with multiple GPUs |
| 12:30 am | 13:30 pm | Andrei Tatarinov,Tim Tcheblokov, Far Cry 4, Assassin's Creed Unity and War Thunder: Spicing up PC graphics with GameWorks |
| 14:00 pm | 15:00 pm | Nathan Reed, VR Direct: How NVIDIA Technology Is Improving The VR Experience |
| 15:30 pm | 16:30 pm | Alexey Panteleev, VXGI: dynamic global illumination for games |
| 17:00 pm | 18:00 pm | Jeffrey Kiel, Russ Kerschner, Brighter and faster graphics with NVIDIA Nsight Visual Studio Edition 4.5™ and beyond |



# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers



# Direct3D 12

- Latest high-performance graphics API
- Low-level model, even more direct
- Works across all Microsoft Platforms
- Supported by excellent tools
- Supports top PC hardware vendors

# DirectX 12 Features

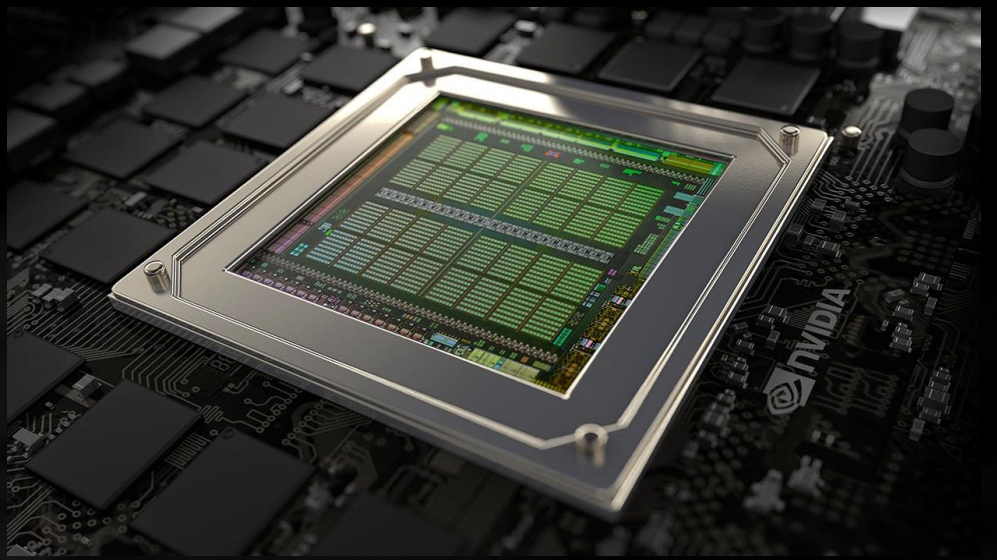
- New API is parallelizable for rendering on multicore CPUs
- Reduced API overhead for single-core work
- More nimble resource binding model using indexing
- More efficient data management/transfer model
- More explicit work scheduling model
- New Hardware Features

- Demos in station at the Microsoft Booth of API and tools



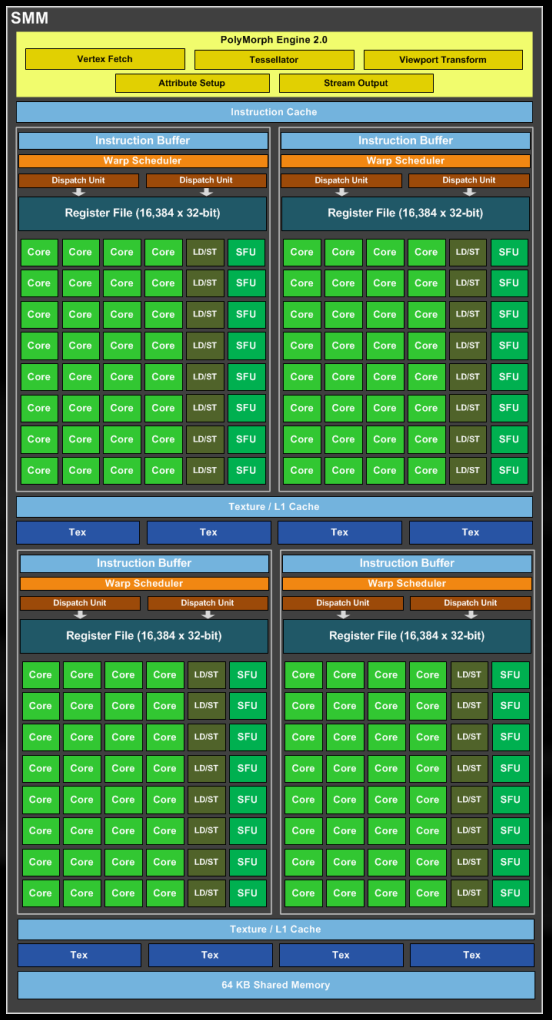
# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers



# Maxwell and its architectural Goals

- New architecture for improved effiency still on a 28nm process
- Massively improved perf / watt
- Focus on new graphics features
  - Real-time GI for rich dynamic scenes
  - Higher quality, programmable AA
  - Working set management
  - Scalable 2D graphics acceleration
  - Create the best platform for DirectX 12



# Maxwell and its architectural Goals

- New architecture for improved effiency still on a 28nm process
- Massively improved perf / watt
- Focus on new graphics features
  - Real-time GI for rich dynamic scenes
  - Higher quality, programmable AA
  - Working set management
  - Scalable 2D graphics acceleration
  - Create the best platform for DirectX 12

| | 680 | 780 | 980 |
|---|---|---|---|
| TFLOPS | 3 | 4 | 5 |
| MEMORY | 2GB | 3GB | 4GB |
| PERFORMANCE | 1 | 1.5 | 2 |
| POWER | 195W | 250W | 165W |
| GFLOPS / WATT | 15 | 15 | 30 |



# Maxwell and its architectural Goals

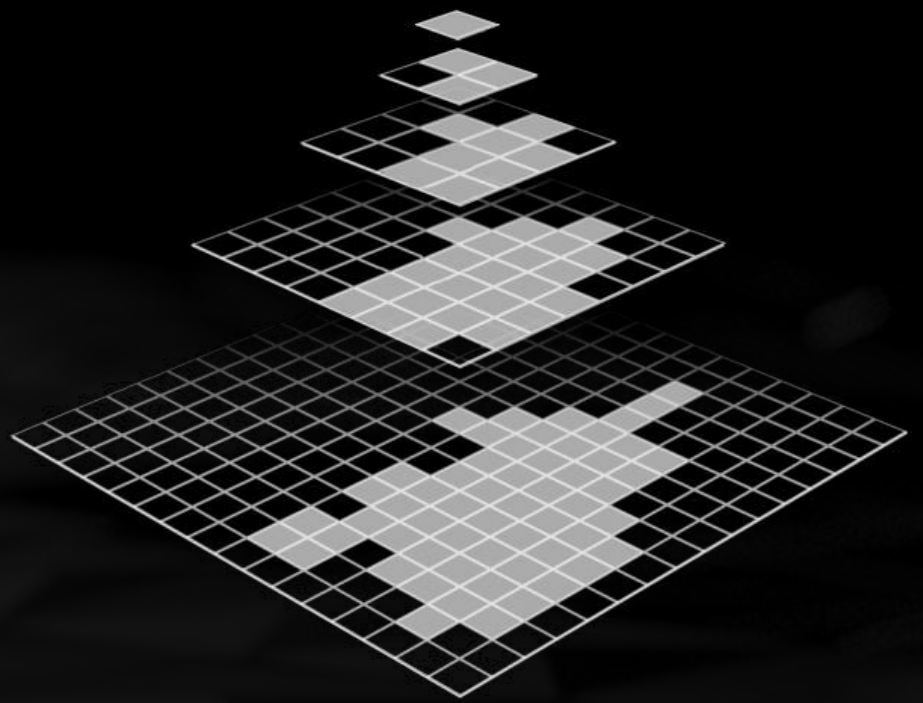
- New architecture for improved effiency still on a 28nm process
- Massively improved perf / watt
- Focus on new graphics features
  - Real-time GI for rich dynamic scenes
  - Higher quality, programmable AA
  - Working set management
  - Scalable 2D graphics acceleration
  - Create the best platform for DirectX 12

# Maxwell and its architectural Goals

- New architecture for improved effiency still on a 28nm process
- Massively improved perf / watt
- Focus on new graphics features
  - Real-time GI for rich dynamic scenes
  - Higher quality, programmable AA
  - Working set management
  - Scalable 2D graphics acceleration
  - Create the best platform for DirectX 12

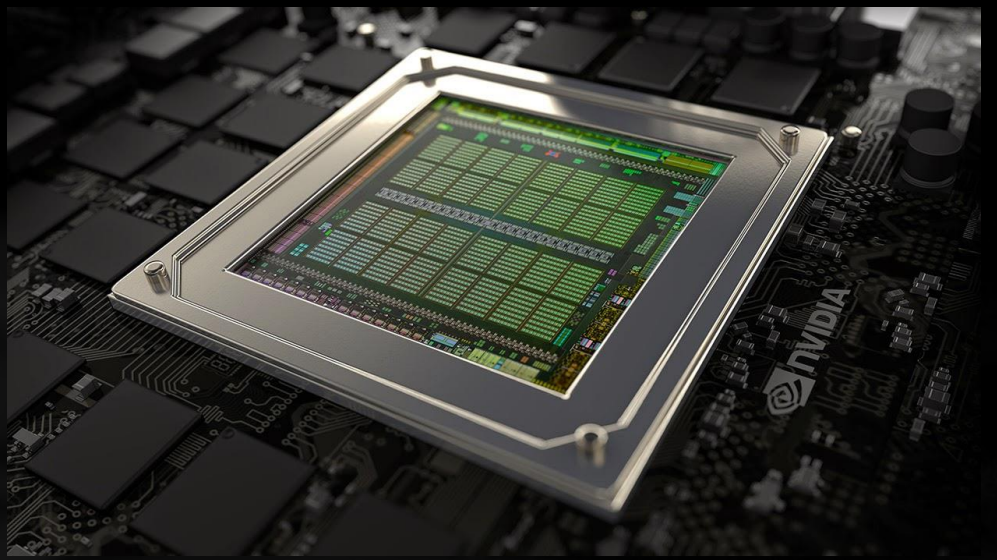
# Maxwell and its architectural Goals

- New architecture for improved effiency still on a 28nm process
- Massively improved perf / watt
- Focus on new graphics features
  - Real-time GI for rich dynamic scenes
  - Higher quality, programmable AA
  - Working set management
  - Scalable 2D graphics acceleration
  - Create the best platform for DirectX 12



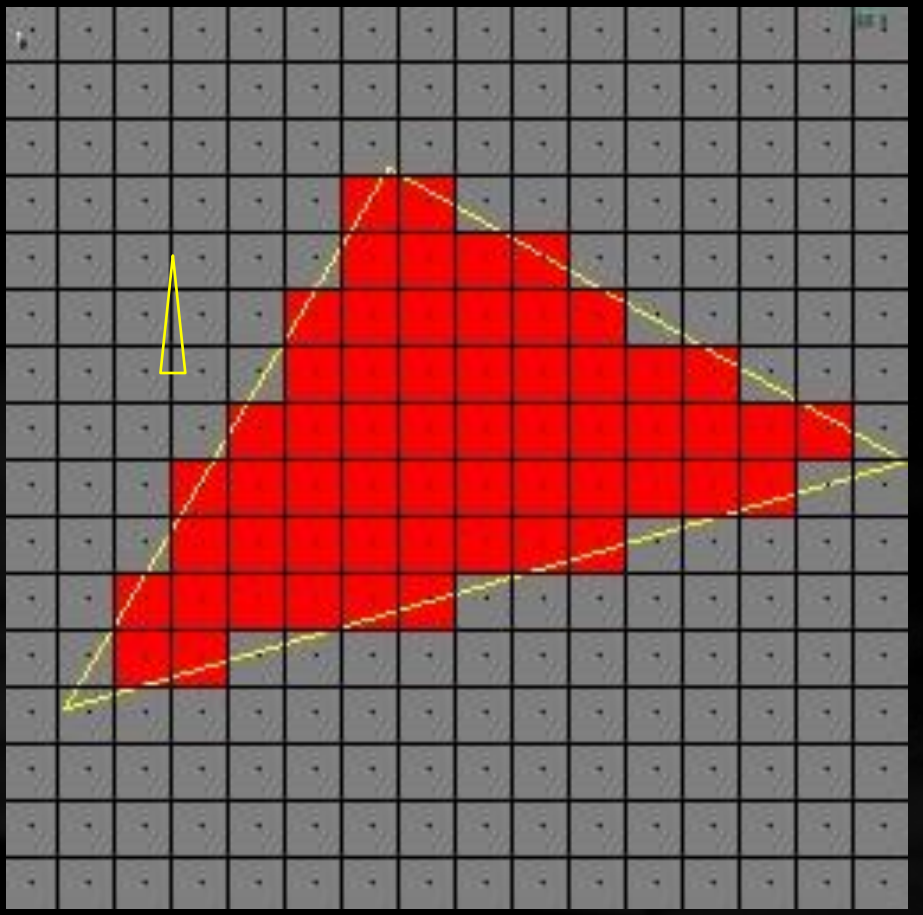
# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers



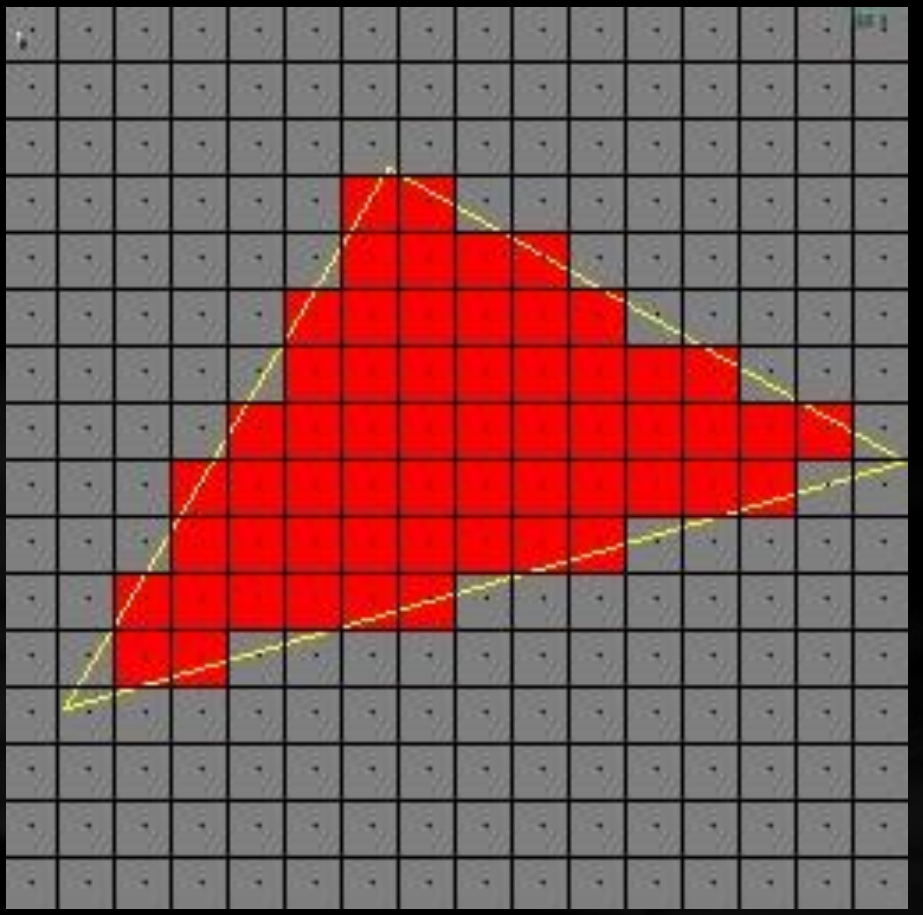
# Standard Rasterization Limitations

- Rasterization can't easily create data-structures
  - Drops sub-pixel triangles
- Data-structures used in later compute passes
  - E.g. Ray-Tracing



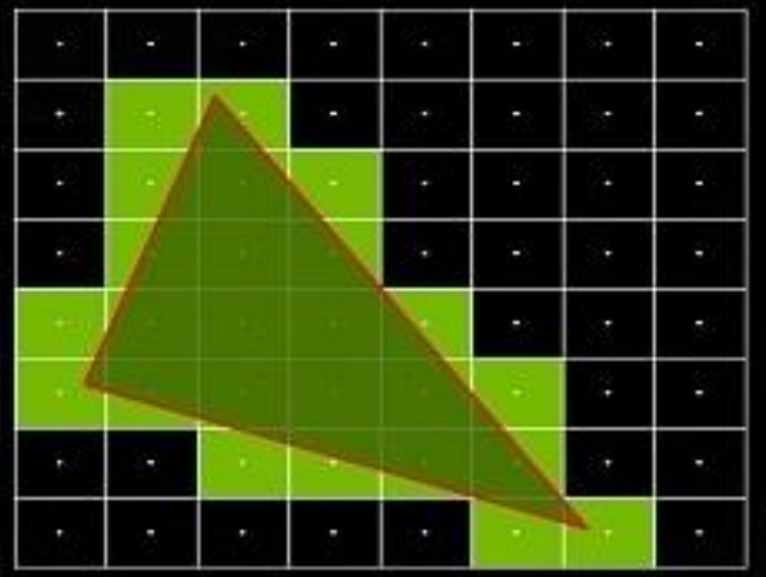
# Standard Rasterization Limitations

- Rasterization can’t easily create data-structures
  - Drops sub-pixel triangles
- Data-structures used in later compute passes
  - E.g. Ray-Tracing



# Conservative Rasterization

- Draws all pixels a triangle touches
  - Different Tiers – see DX spec
- Possible before through GS trick but relatively slow
  - See J. Hasselgren et. Al, “Conservative Rasterization“, GPU Gems 2
- Now we can use rasterization do implement some nice techniques!

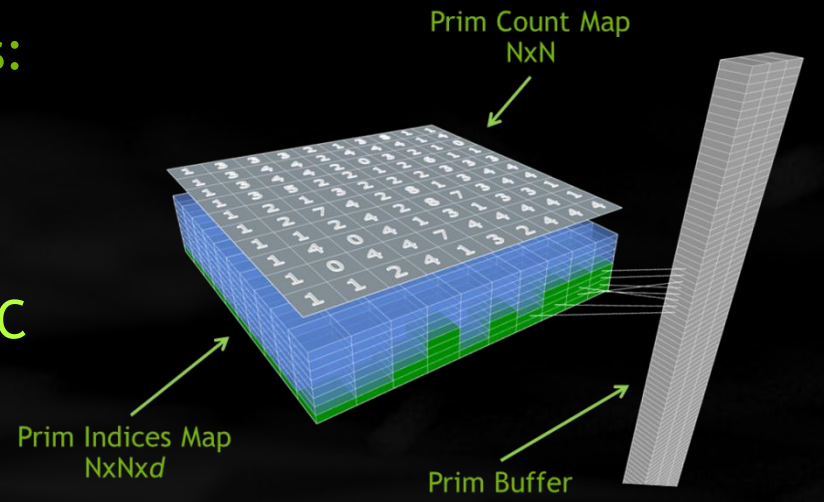


# Conservative Rasterization Usecases

- C. Wymann et. al, “Frustum-Traced Raster Shadows: Revisiting Irregular Z-Buffers“, I3D 2015
- J. Story “Hybrid Ray-Traced Shadows“, D3D Day GDC 2015

# Conservative Rasterization Usecases

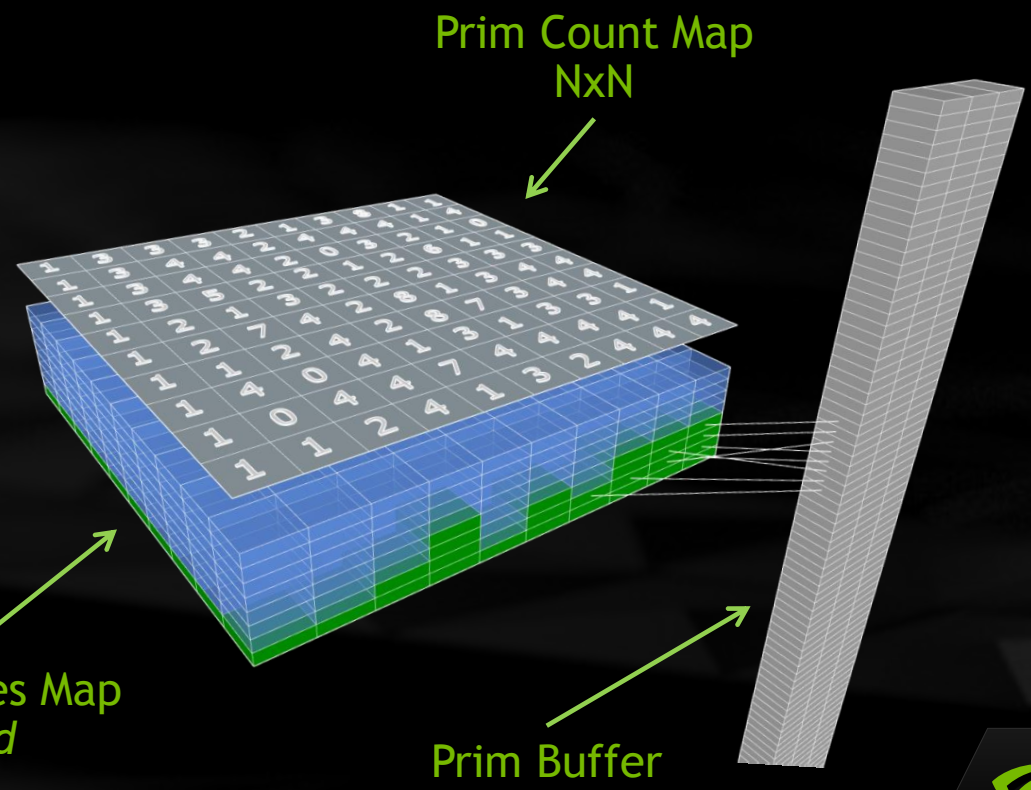
- C. Wymann et. al, “Frustum-Traced Raster Shadows: Revisiting Irregular Z-Buffers“, I3D 2015
- J. Story “Hybrid Ray-Traced Shadows“, D3D Day GDC 2015



# Hybrid Raytraced Shadows

## using a NxNx*d* primitive map

- See J. Story, “Hybrid Ray-Traced Shadows“, D3D Day GDC 2015
- Prim Buffer – Triangle vertices
- Prim Indices Map – Prim buffer indices of triangles
- Prim Count Map – # of tris per texel
- Raytrace triangles in a later pass

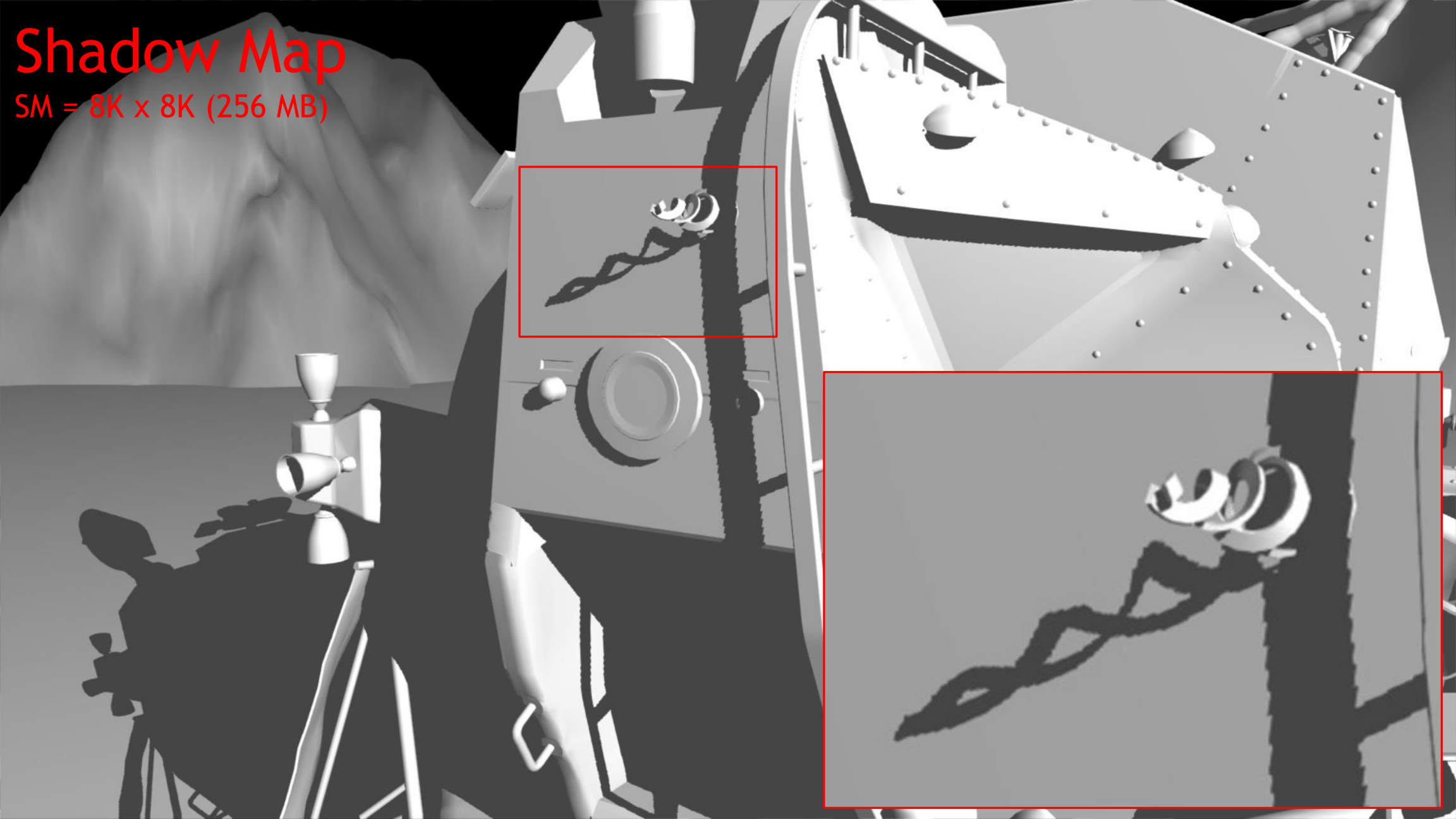


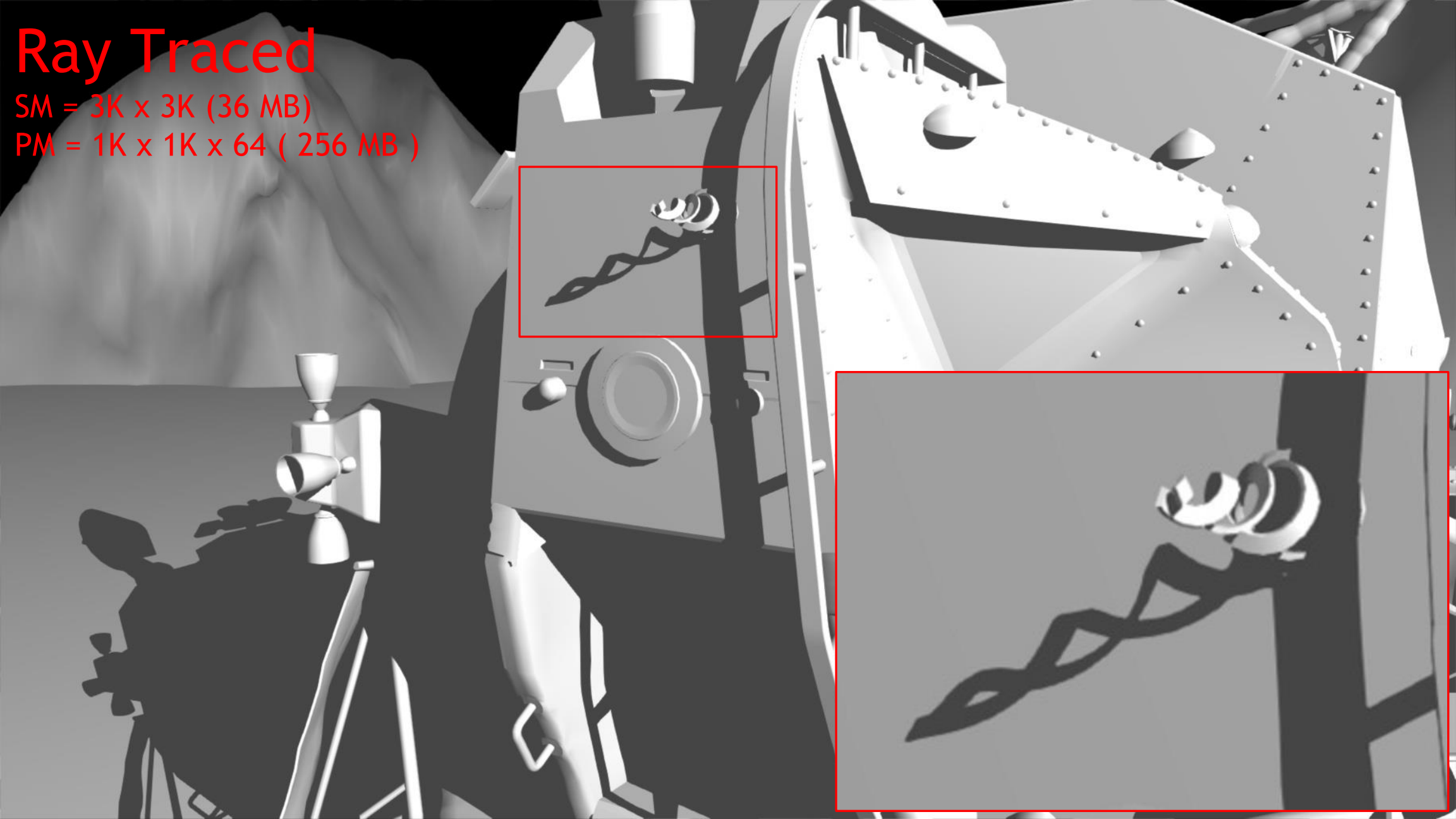
# Raytraced Shadows Demo



# Shadow Map

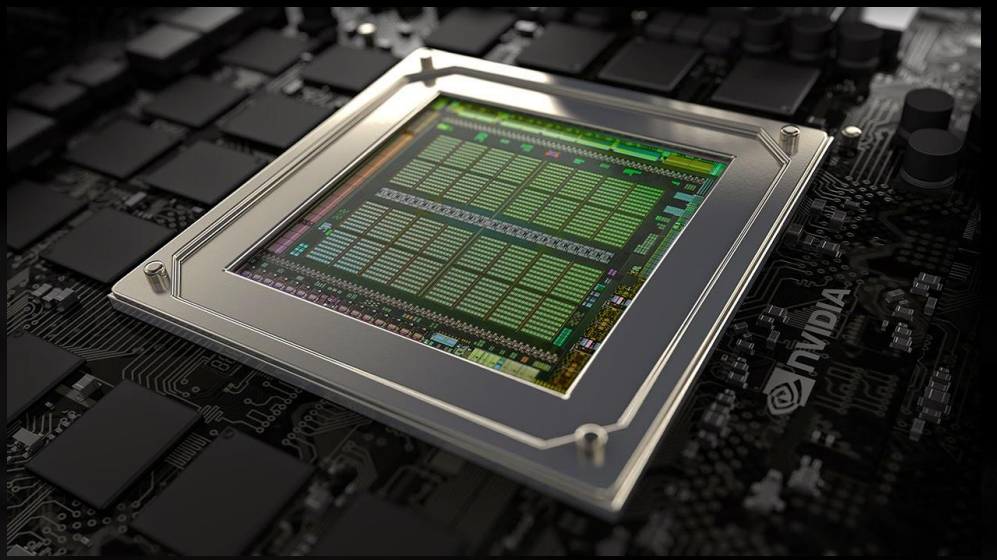
SM = 8K x 8K (256 MB)




Ray Traced
SM = 3K x 3K (36 MB)
PM = 1K x 1K x 64 ( 256 MB )

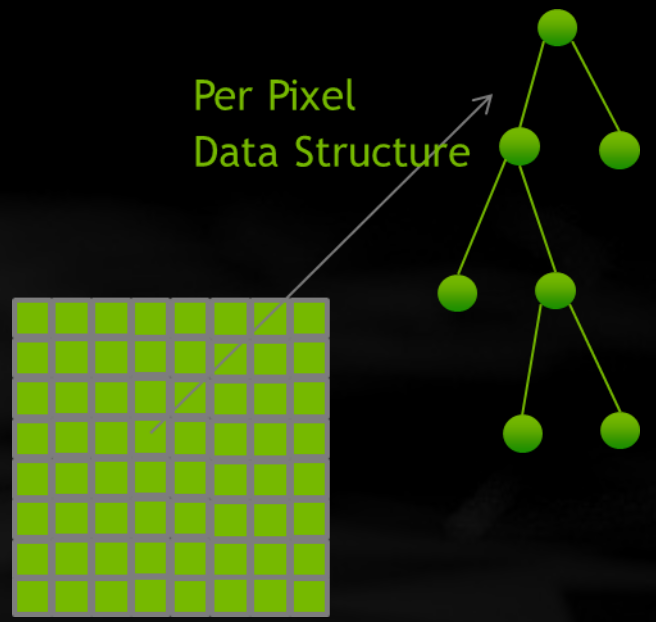
# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers



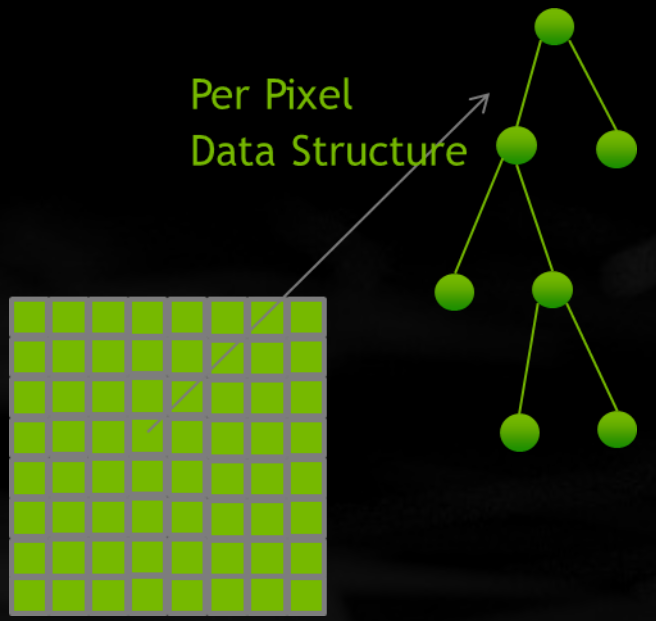
# UAV Race Condition Issue

- Pixel shader writes to UAVs are unordered
  - Can‘t guarantee determinism
- Can‘t do a number of things
  - Programmable Blending
  - Smart OIT implementations
  - Arbitray g-buffer data packing
  - Other per-pixel-data structures



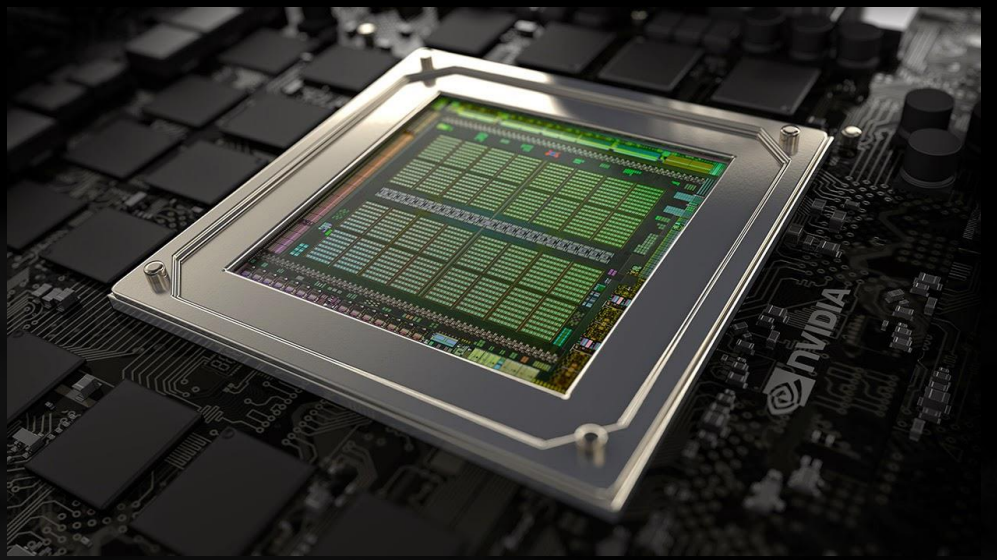
# Raster Order Views (ROVs)

- ROVs guarantee ordering
- Ordering doesn‘t come for free
  - Depth complexity affects performance
- Always compare with alternative implementations
  - Advanced Blending Ops
  - Atomics



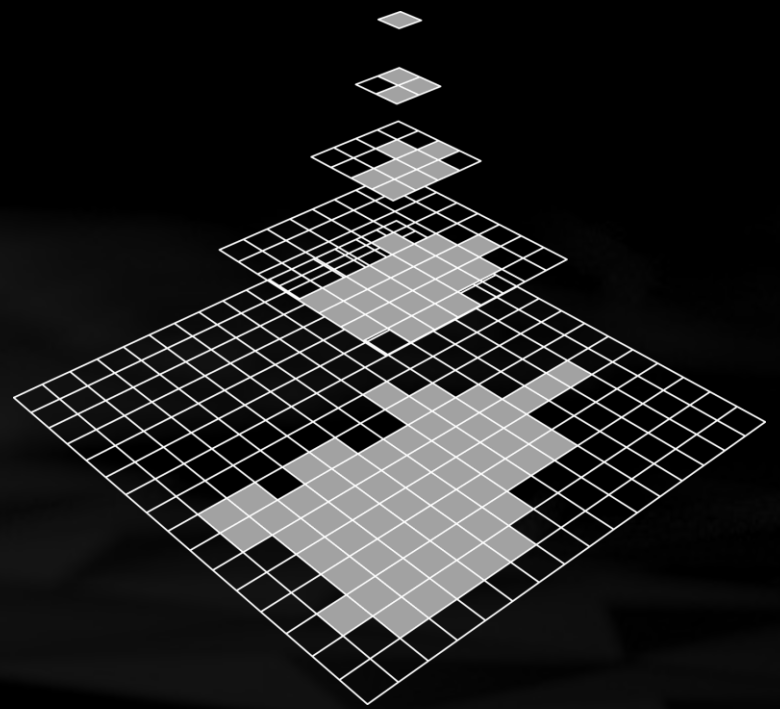
# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers



# DX12 Tiled Resources

- Full support for tiled 3D Textures/Arrays
  - On top of what DX11.2 provides
- Recap Tiled Resources
  - Enable fine grained working set management
  - Texture defined as set of tiles
  - Memory for tiles allocated  separately



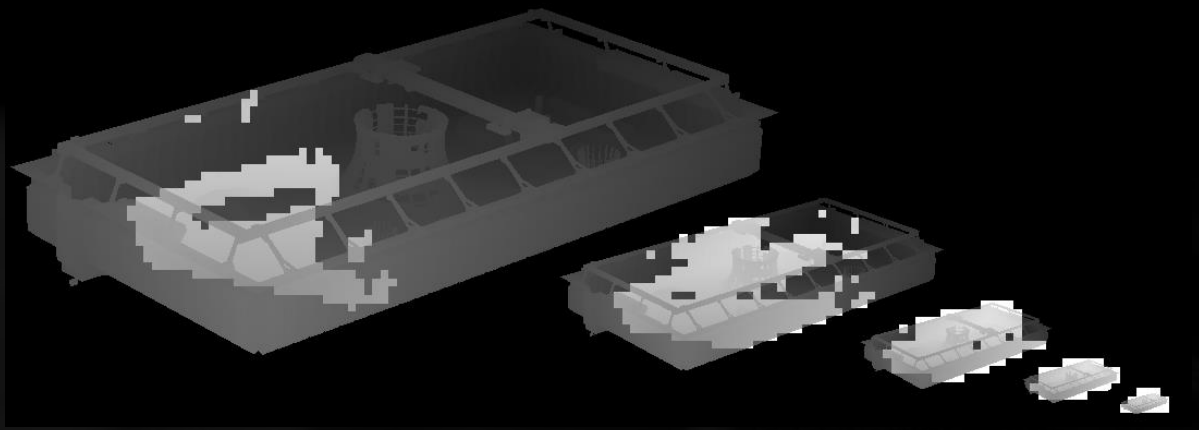
# Tiled Resources applications

- Fine-grained working set management
  - Texture streaming/ Clipmaps
- Variable resolution resources
  - Adaptive shadow maps
  - Sparse multi-resolution rendering
- Sparse representation
  - Voxel grids
  - Simulation – physics, path finding



# Tiled Resources applications

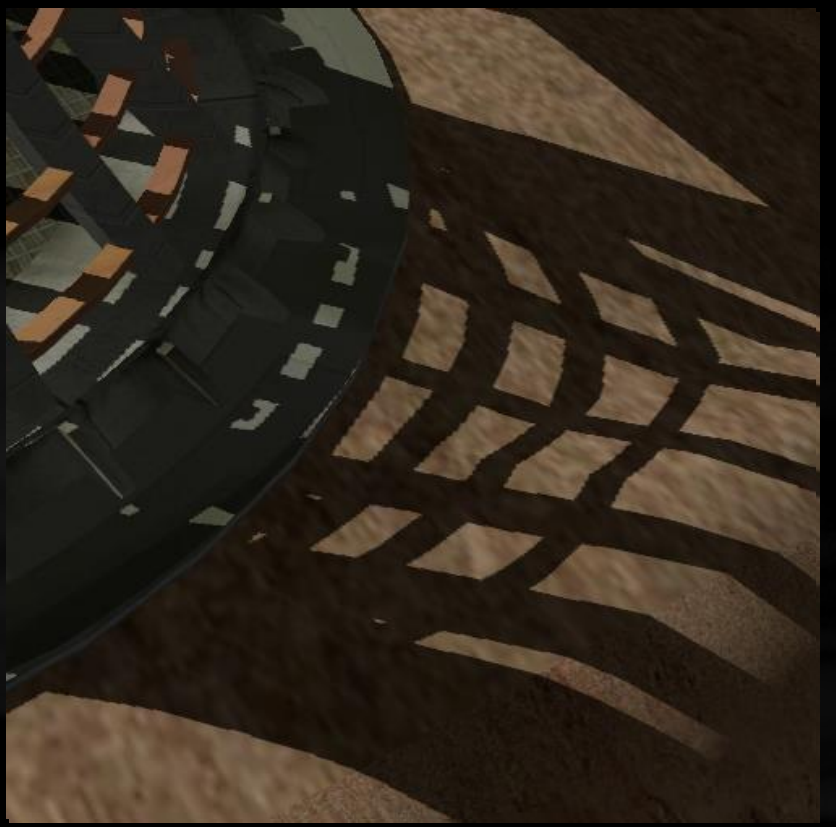
- Fine-grained working set management
  - Texture streaming/ Clipmaps
- Variable resolution resources
  - Adaptive shadow maps
  - Sparse multi-resolution rendering
- Sparse representation
  - Voxel grids
  - Simulation – physics, path finding





# Tiled Resources applications

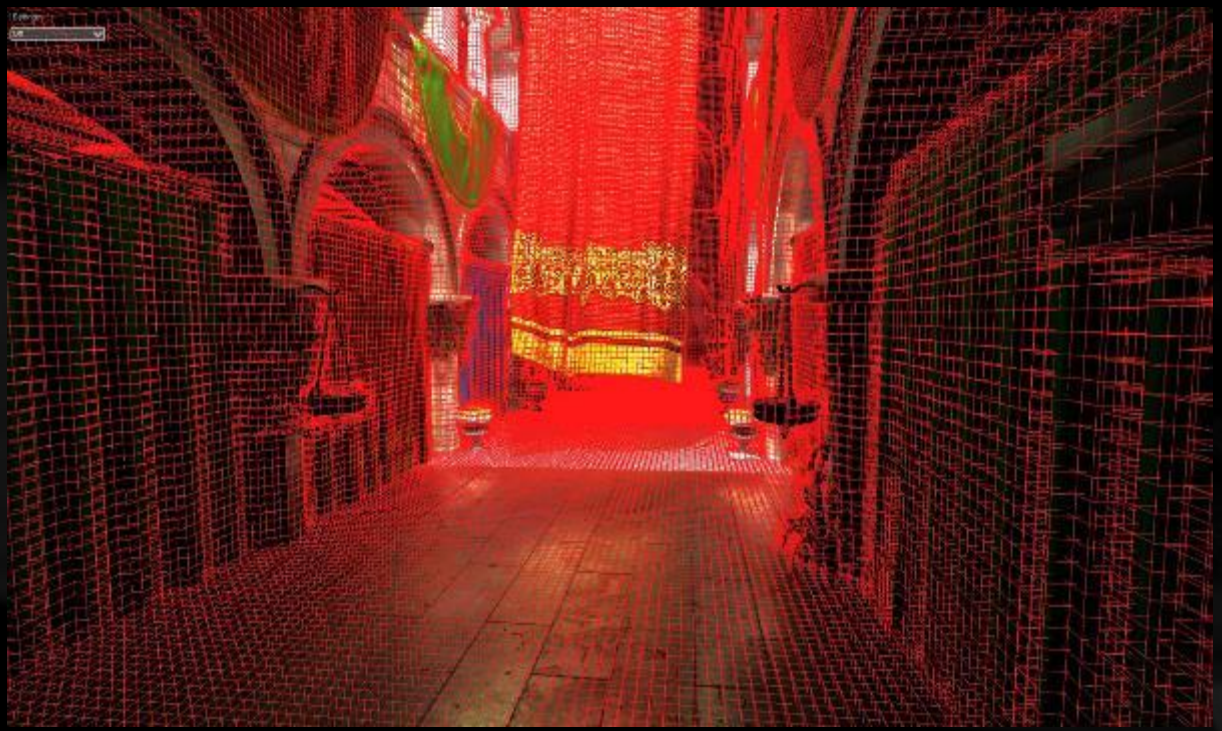
- Fine-grained working set management
  - Texture streaming/ Clipmaps
- Variable resolution resources
  - Adaptive shadow maps
  - Sparse multi-resolution rendering
- Sparse representation
  - Voxel grids
  - Simulation – physics, path finding





# Tiled Resources applications

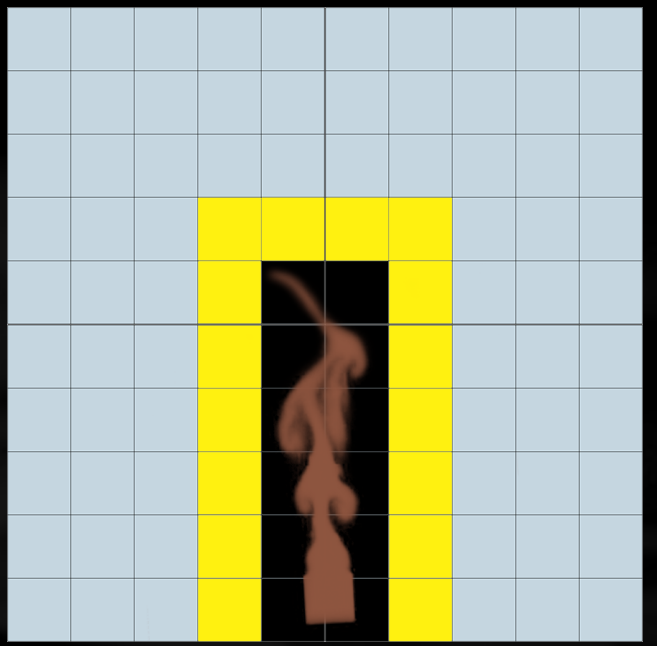
- Fine-grained working set management
  - Texture streaming/ Clipmaps
- Variable resolution resources
  - Adaptive shadow maps
  - Sparse multi-resolution rendering
- Sparse representation
  - Voxel grids
  - Simulation – physics, path finding





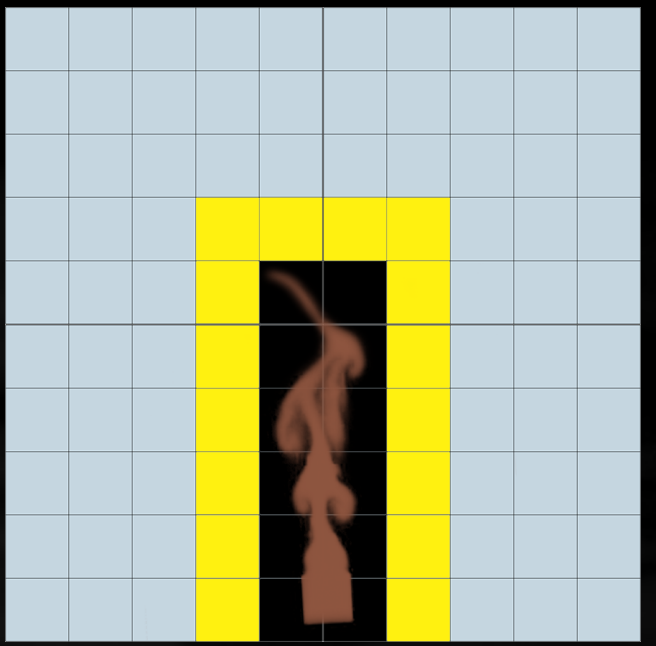
# Tiled Resources applications

- Fine-grained working set management
  - Texture streaming/ Clipmaps
- Variable resolution resources
  - Adaptive shadow maps
  - Sparse multi-resolution rendering
- Sparse representation
  - Voxel grids
  - Simulation – physics, path finding



# Sparse Fluid Simulation

- Uses tiled resources to only simulate/store grid cells that contain fluid
- Save computation time and memory
- See Alex Dunn, ”Sparse Fluid Simulation in DirectX”, D3D Day GDC 2015

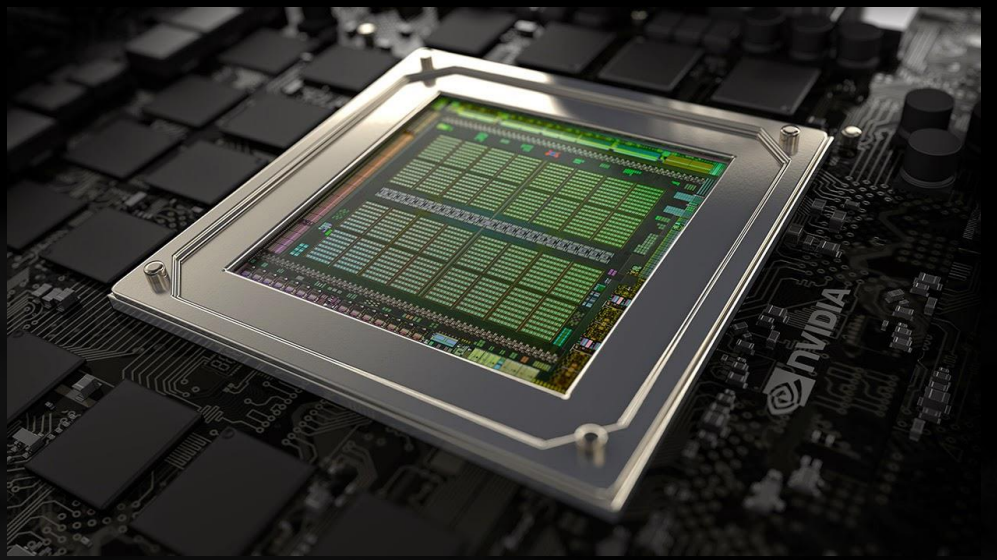


# Sparse Fluid Demo



# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers



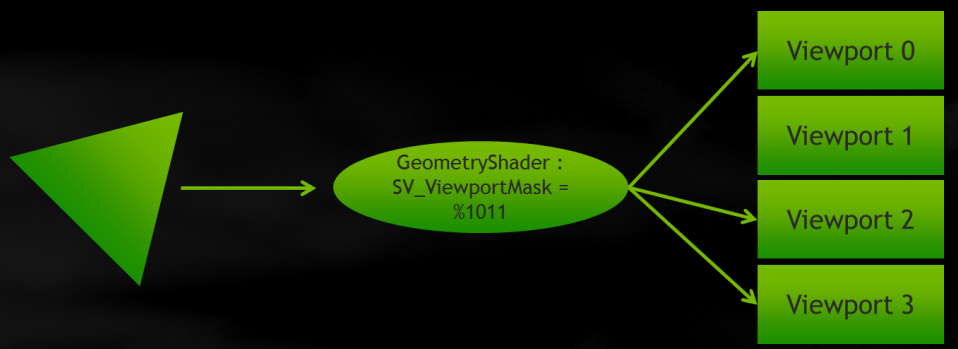
# Geometry Shader Challenges

- Significant overhead even for pass-through cases
- Significant overhead for viewport selection
- Significant amplification overhead for multiple viewports



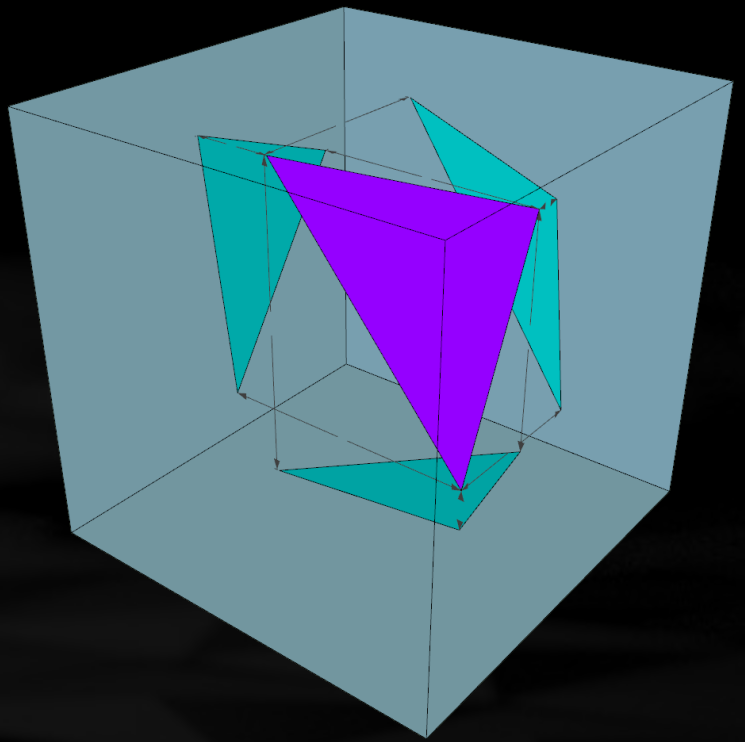
# Multi-Projection Acceleration

- Fast Geometry shader pass-through
- Fast viewport multi-casting
- Maxwell accelerates:
  - Voxelization
  - Cube-Map Rendering
  - Cascaded shadow maps
  - Multi-resolution rendering



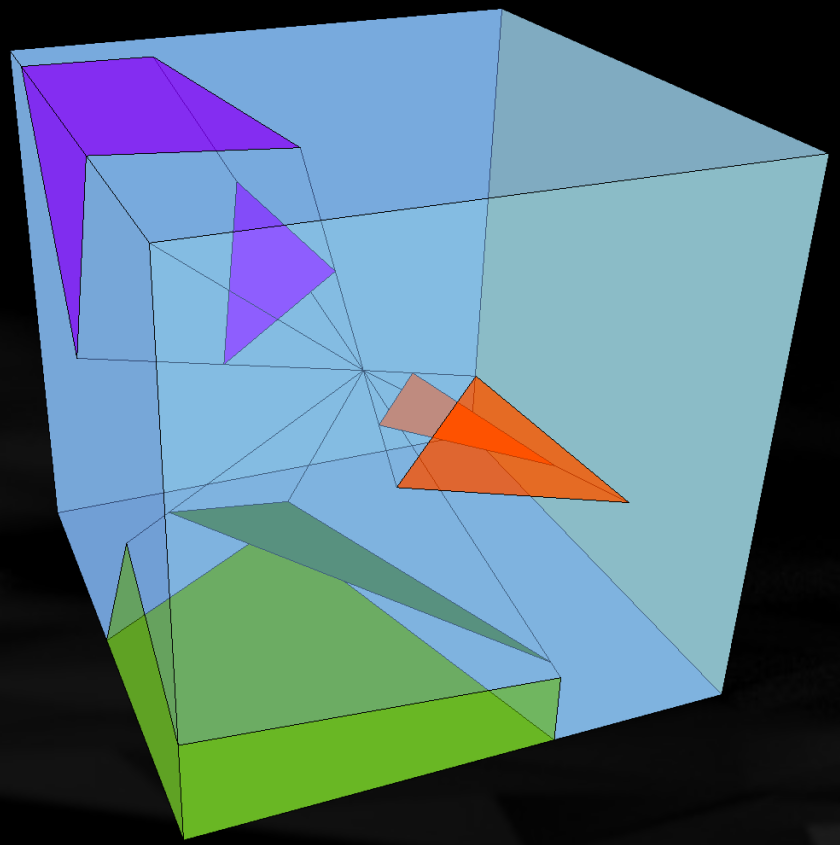
# Multi-Projection Acceleration

- Fast Geometry shader pass-through
- Fast viewport multi-casting
- Maxwell accelerates:
  - Voxelization
  - Cube-Map Rendering
  - Cascaded shadow maps
  - Multi-resolution rendering



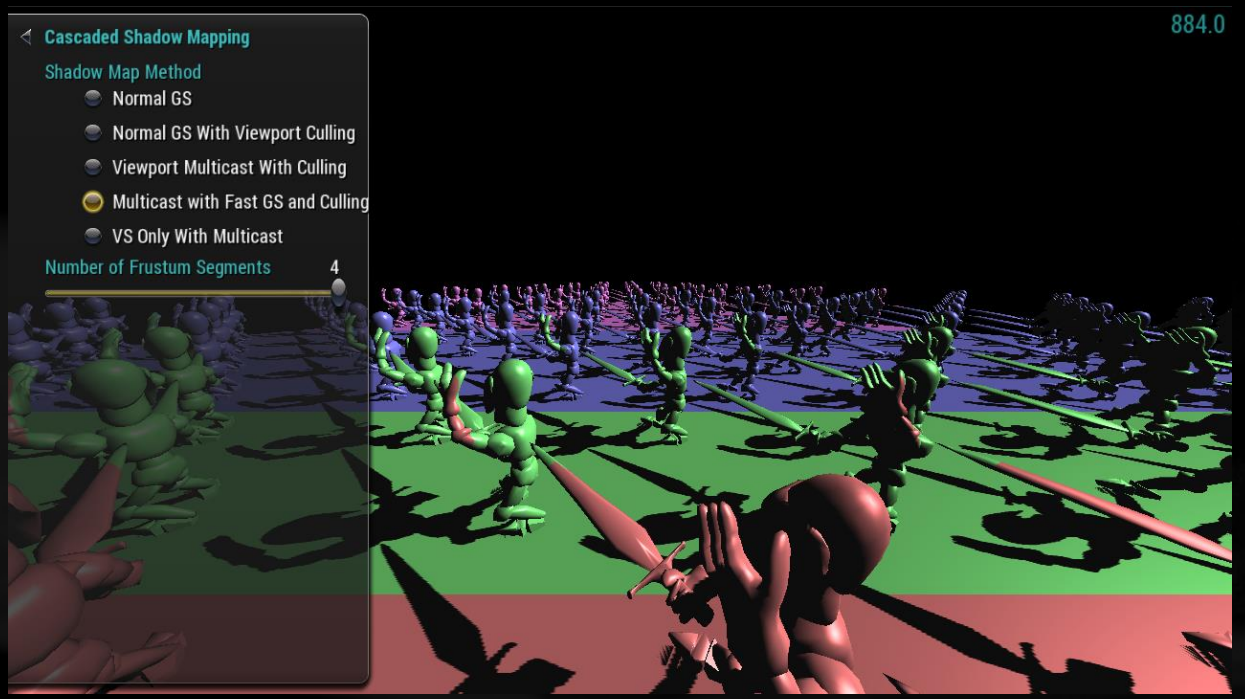
# Multi-Projection Acceleration

- Fast Geometry shader pass-through
- Fast viewport multi-casting
- Maxwell accelerates:
  - Voxelization
  - Cube-Map Rendering
  - Cascaded shadow maps
  - Multi-resolution rendering



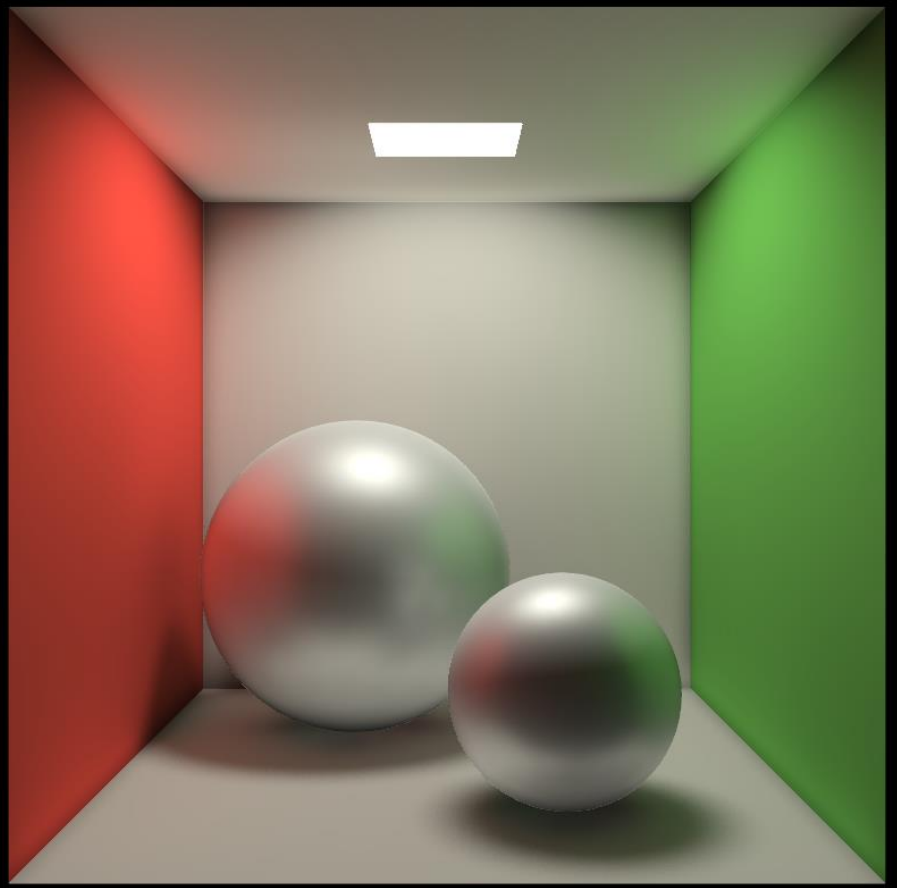
# Multi-Projection Acceleration

- Fast Geometry shader pass-through
- Fast viewport multi-casting
- Maxwell accelerates:
  - Voxelization
  - Cube-Map Rendering
  - Cascaded shadow maps
  - Multi-resolution rendering

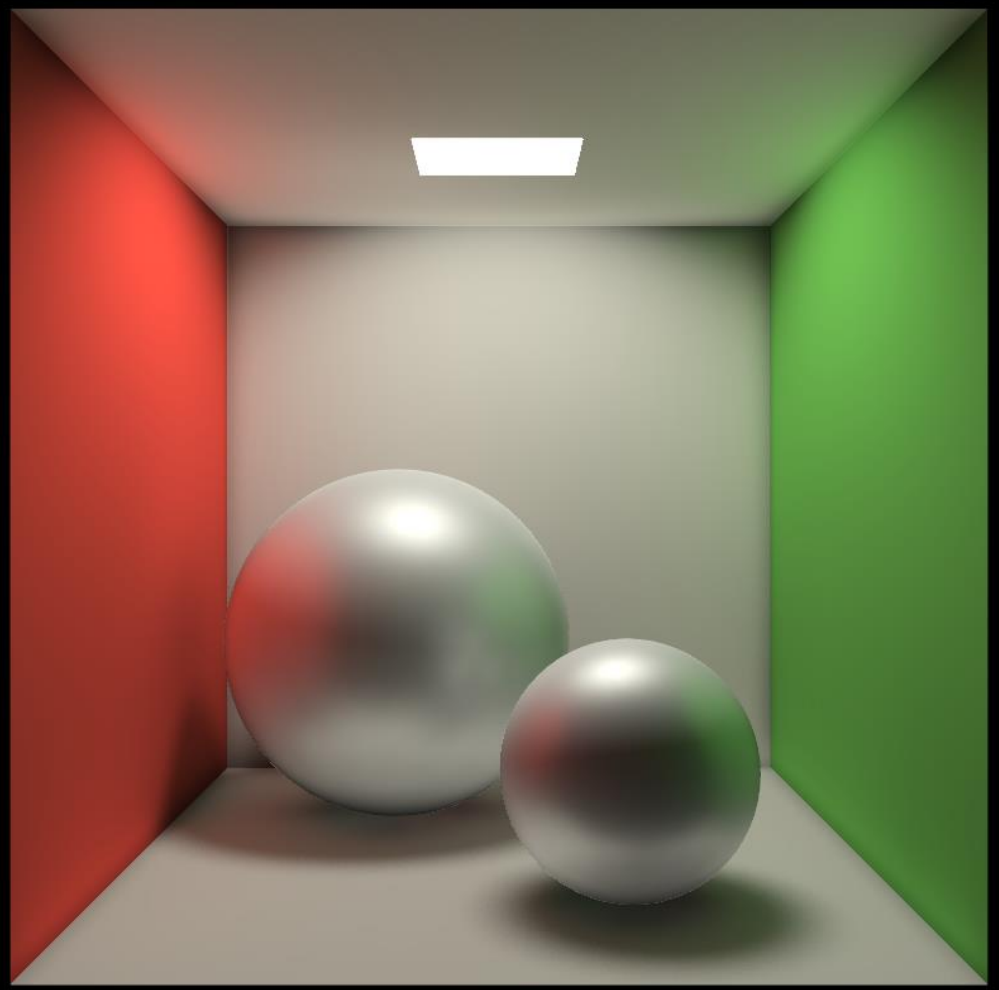


# Voxel Based GI - VXGI

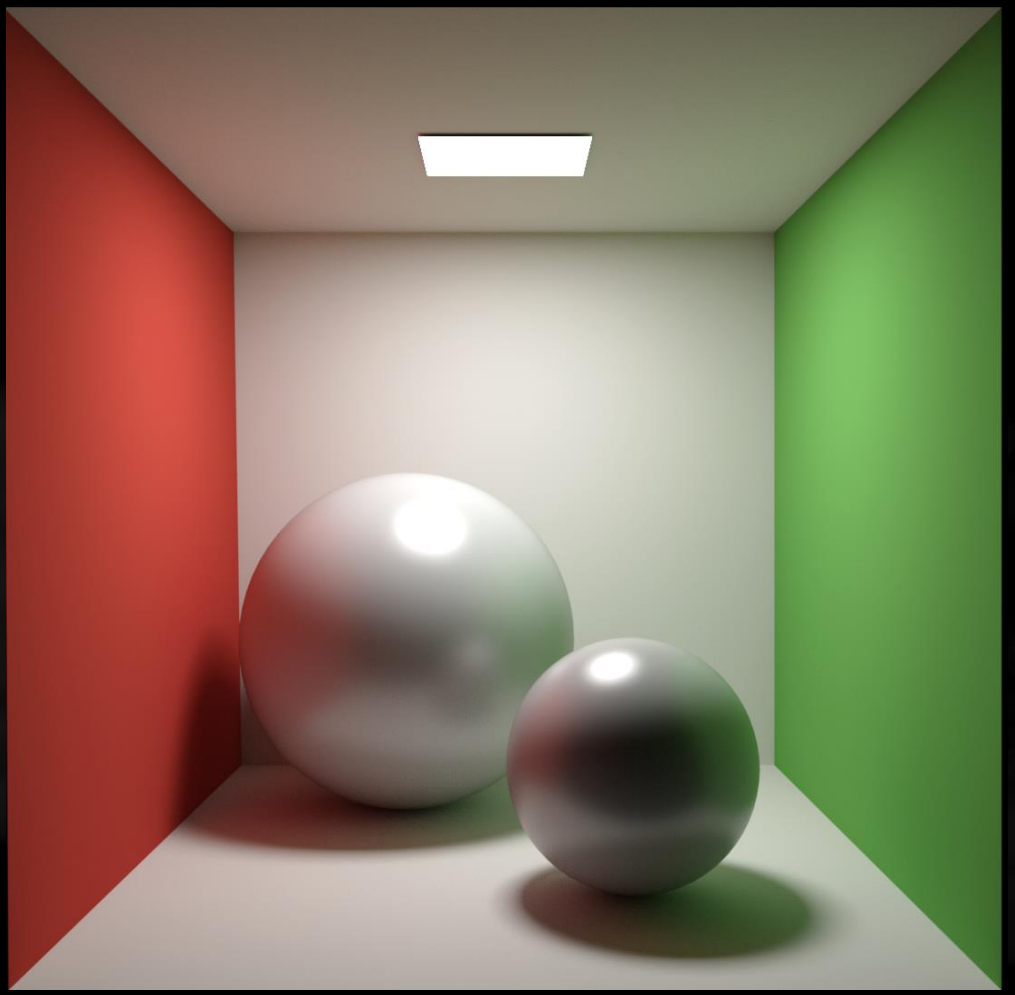
- Uses multi-projection for fast voxelization
- See Alexey Panteleev's talk later today, VXGI: dynamic global illumination for games



# VXGI Result

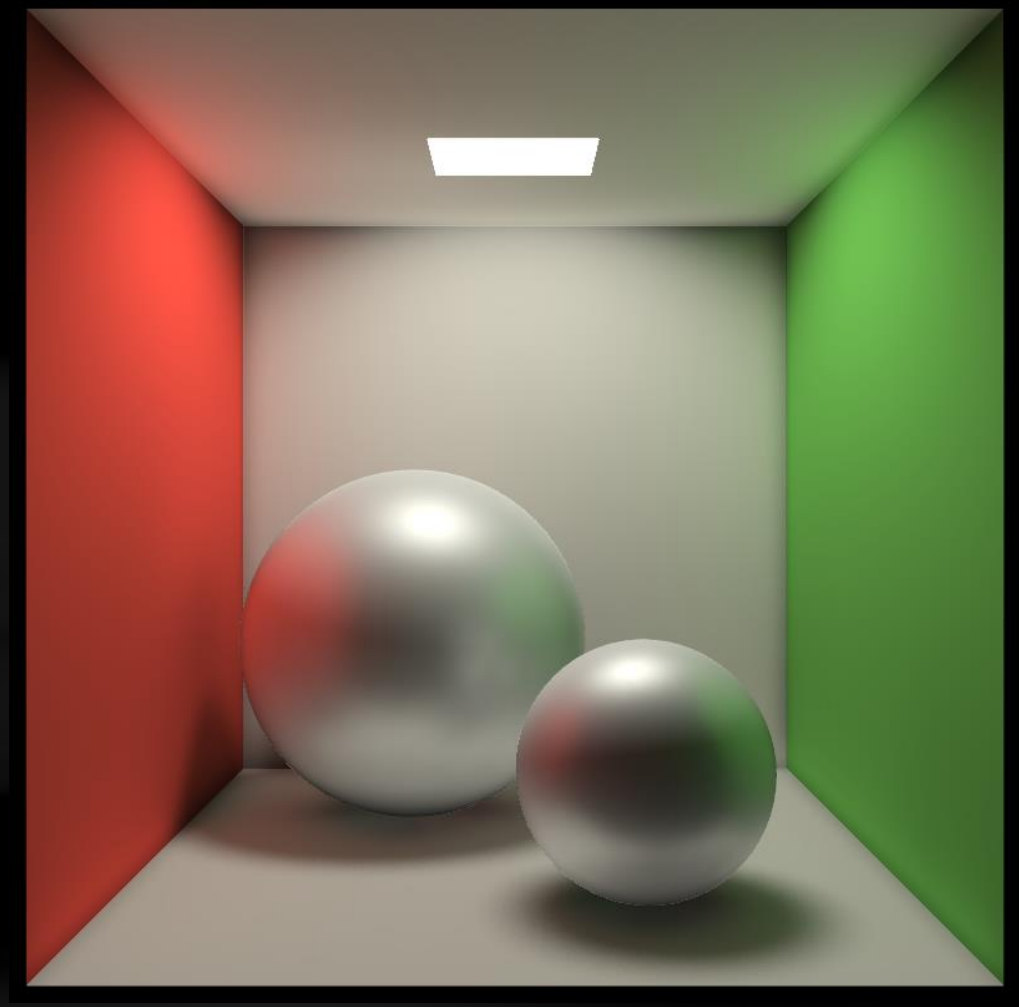


VXGI



Reference rendering with NVIDIA Iray

# VXGI demo



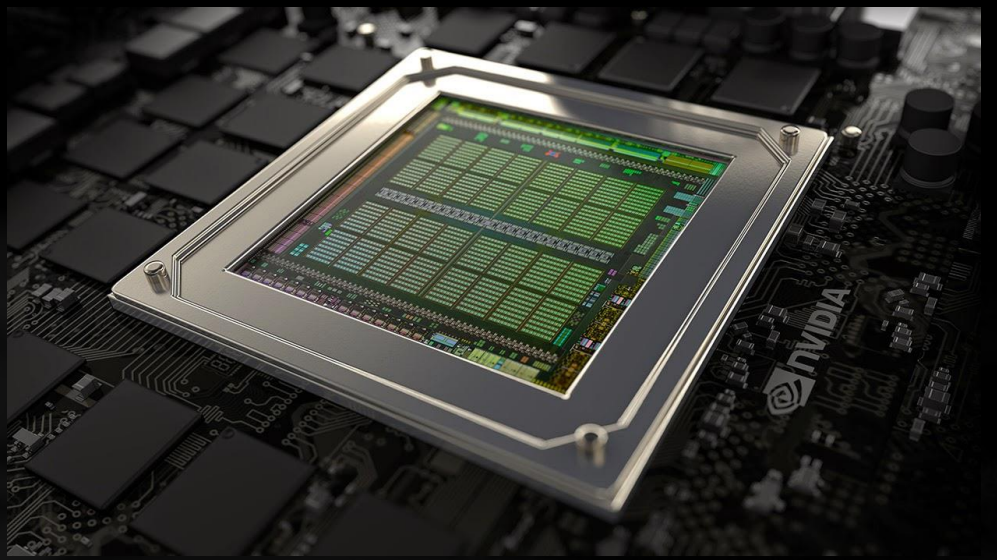
# Multi-Projection API Support

- OpenGL+Android:
  - NV_geometry_shader_passthrough extension for GS pass-through
  - NV_viewport_array2 extension for viewport multicast
  - The extension specs have good shader examples
- DX11/DX12:
  - No explicit API publicly available yet – stay tuned

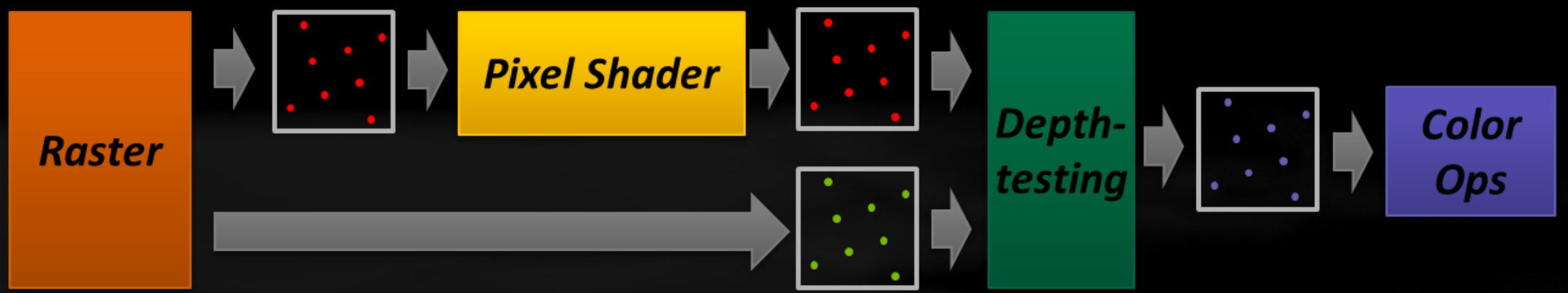


# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers

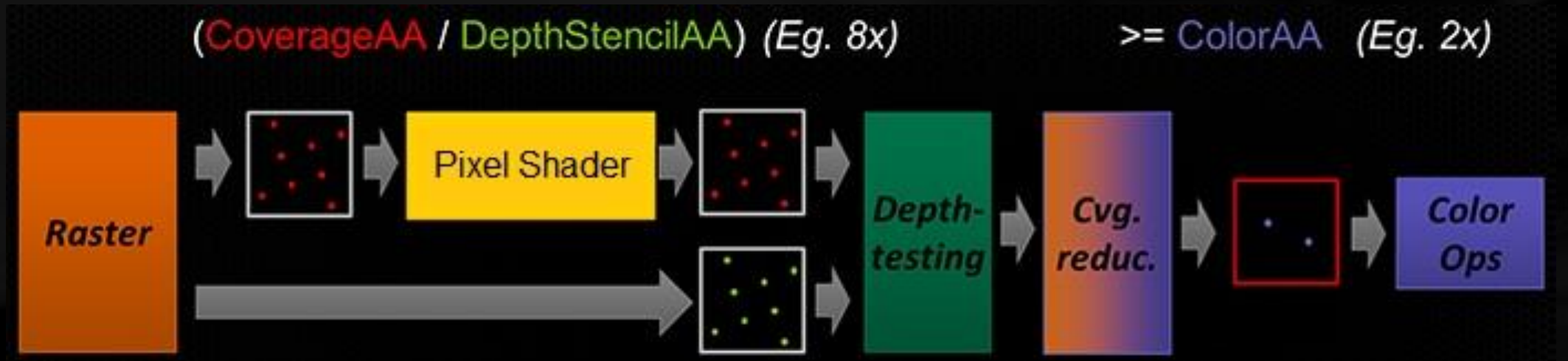


# Quick Multisampling Recap



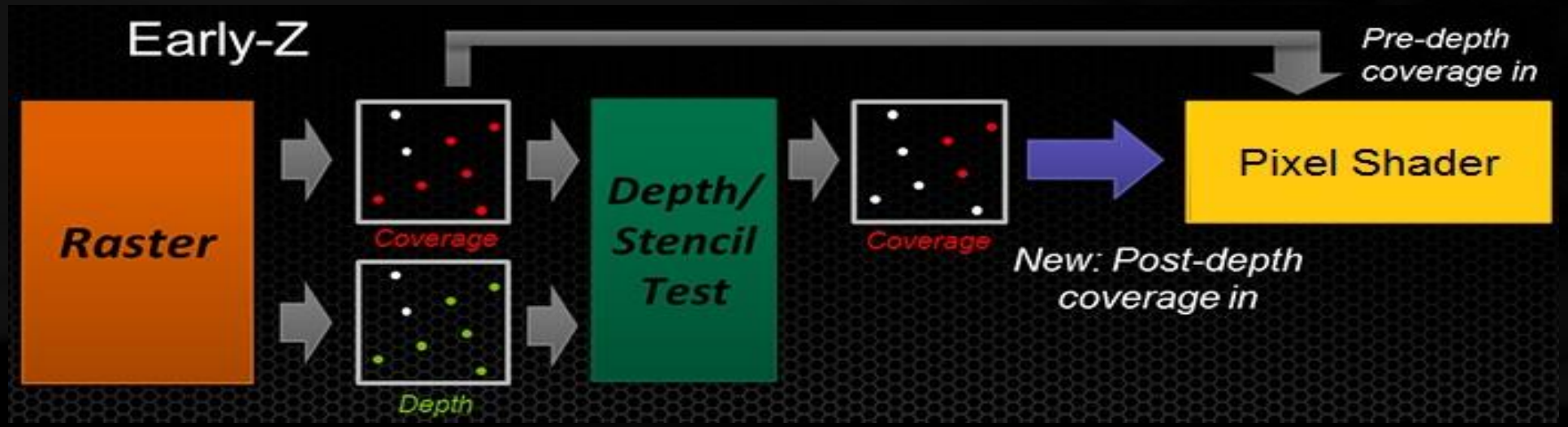
# Target-independent Multisampling

- Decouples visibility rate from color sample rate
- Allows lower color buffer storage cost for custom AA techniques
- Introduces coverage reduction stage



# Post-depth Coverage

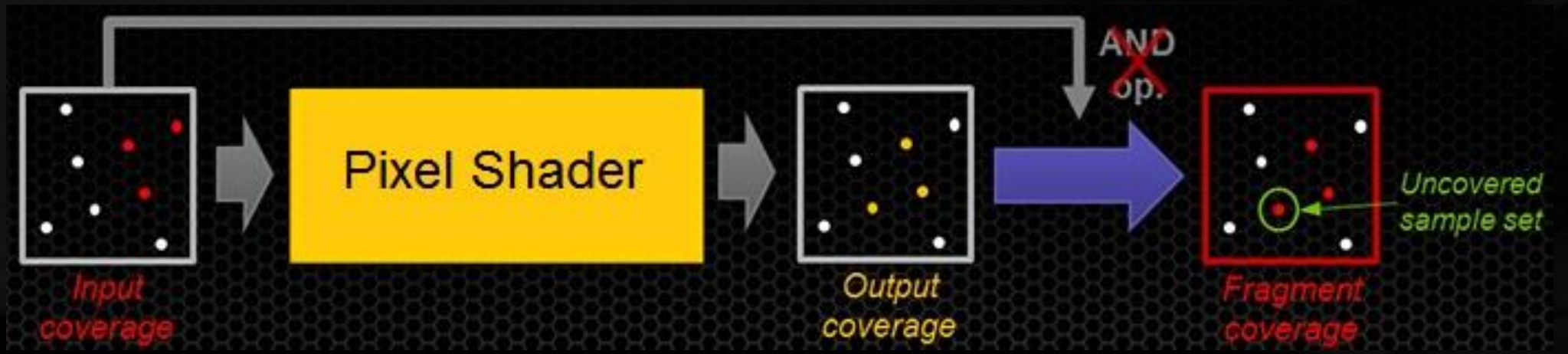
- Pre-Maxwell : Coverage Mask delivered is pre-depth-test coverage
  - No way to get at the post-depth-test coverage
- Maxwell can deliver post-depth-coverage to the pixel shader



# Multisample Coverage Override

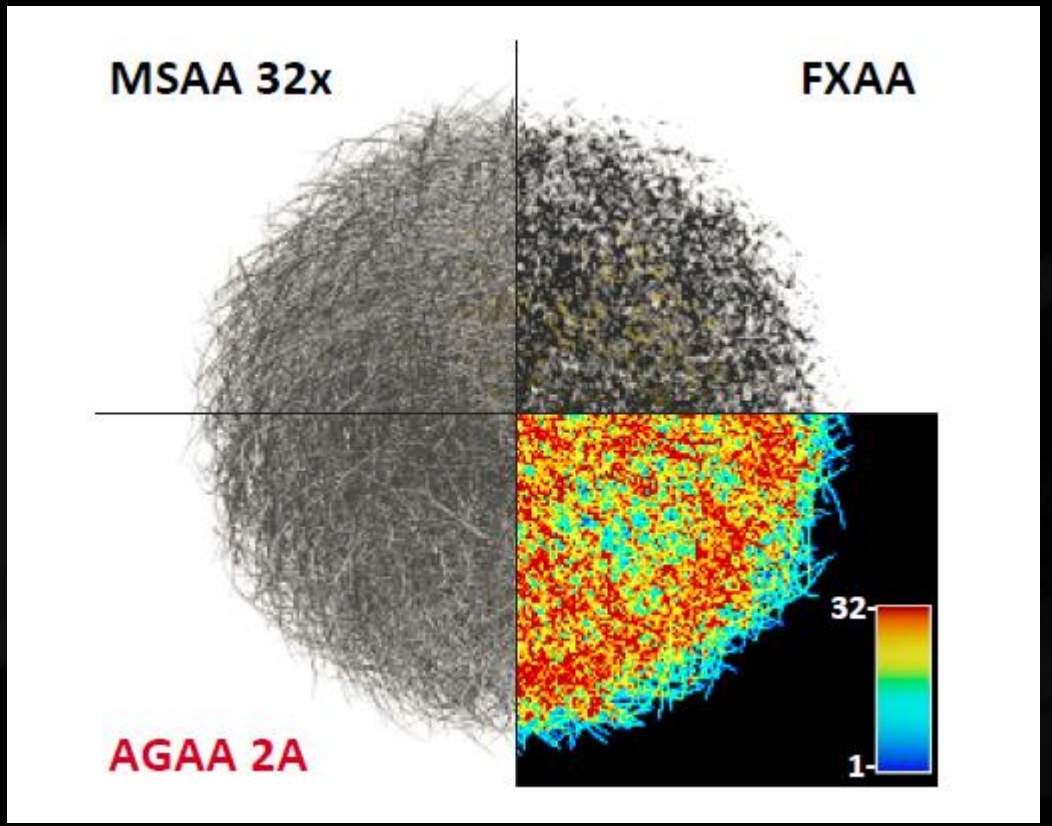
- Pre-Maxwell : Shader can only reduce coverage sample set

- Maxwell can fully override raster-coverage mask

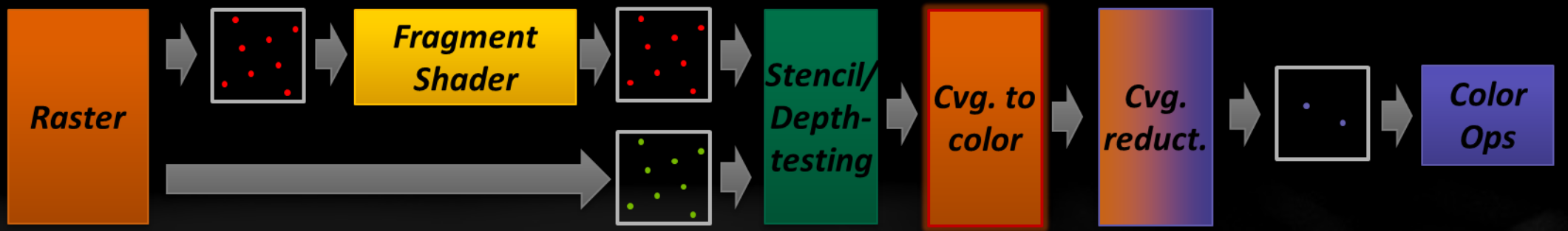


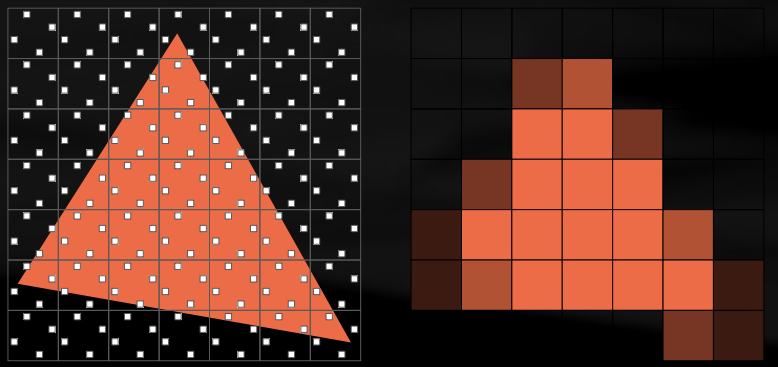
# Aggregate G-Buffer AA demo

- C. Crassin et. al, ”Aggregate G-Buffer Anti-Aliasing”, ID3D 2015
- Uses post depth coverage to only process visible sub-samples
- Uses coverage override to route to right sub-sample cluster
- Other work using Maxwell AA features:
  - E. Enderton et. al, ”Accumulative Anti-Aliasing”, to appear



# MSAA Coverage to Color Conversion



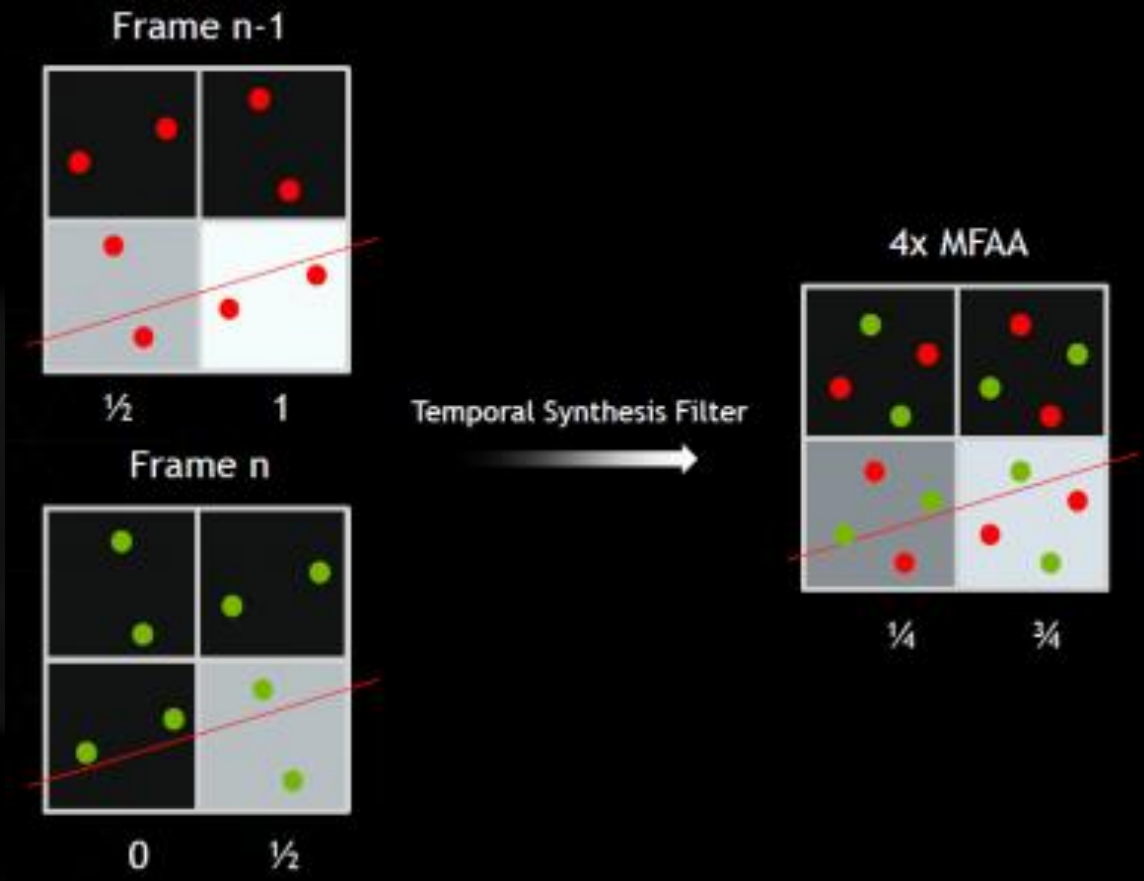



# Programmable Sample Locations

- Sample locations fully programmable
- Foundation for Multi Frame sampled AA
- Interleaved sample positions
  - 16x sample locations can be tiled to a set of pixels

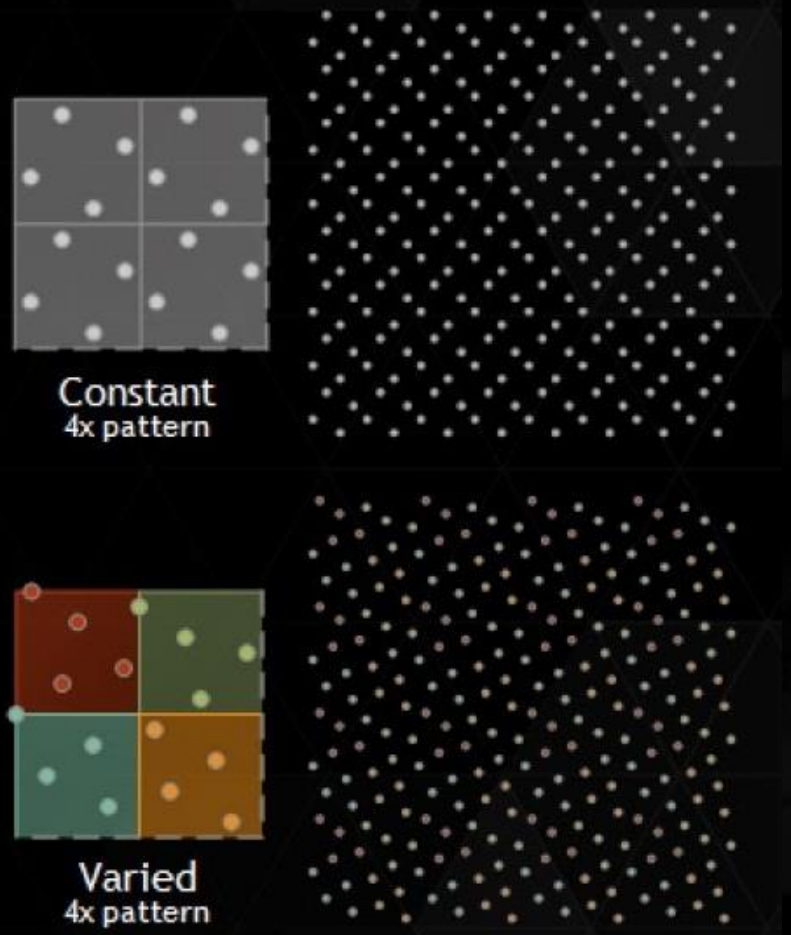
# Programmable Sample Locations

- Sample locations fully programmable
- Foundation for Multi Frame sampled AA
- Interleaved sample positions
  - 16x sample locations can be tiled to a set of pixels



# Programmable Sample Locations

- Sample locations fully programmable
- Foundation for Multi Frame sampled AA
- Interleaved sample positions
  - 16x sample locations can be tiled to a set of pixels



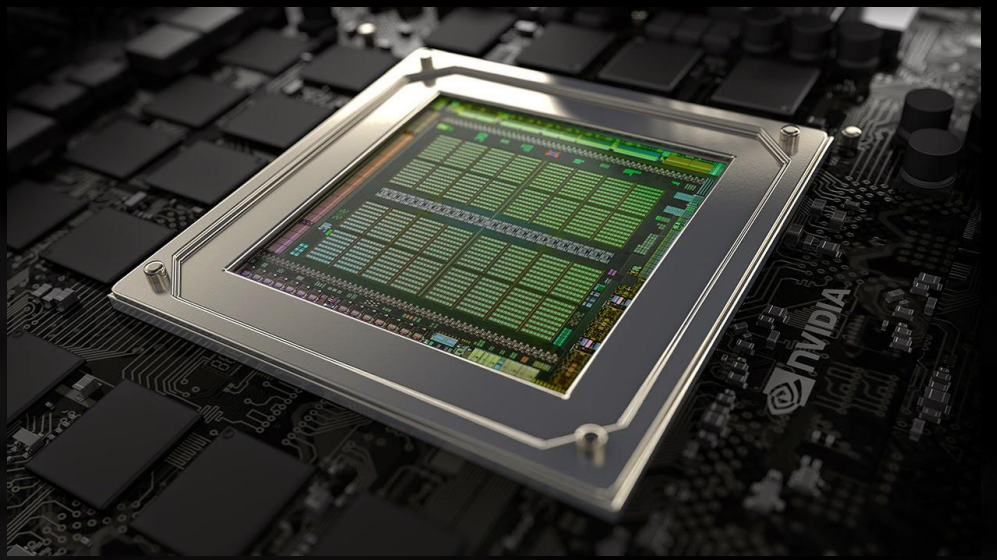
# API support Antialiasing Features

- DirectX 12
  - Target-independent multipsampling
- OpenGL+ Android:
  - Target-independent multisampling control:
    - NV_framebuffer_mixed_samples
    - EXT_raster_multisample
  - Coverage to color conversion: NV_fragment_coverage_to_color
  - Post-depth coverage : EXT_post_depth_coverage
  - Multisample coverage override : NV_sample_mask_override_coverage
  - Programmable sample locations : NV_sample_locations
- NvAPI:
  - Coming soon



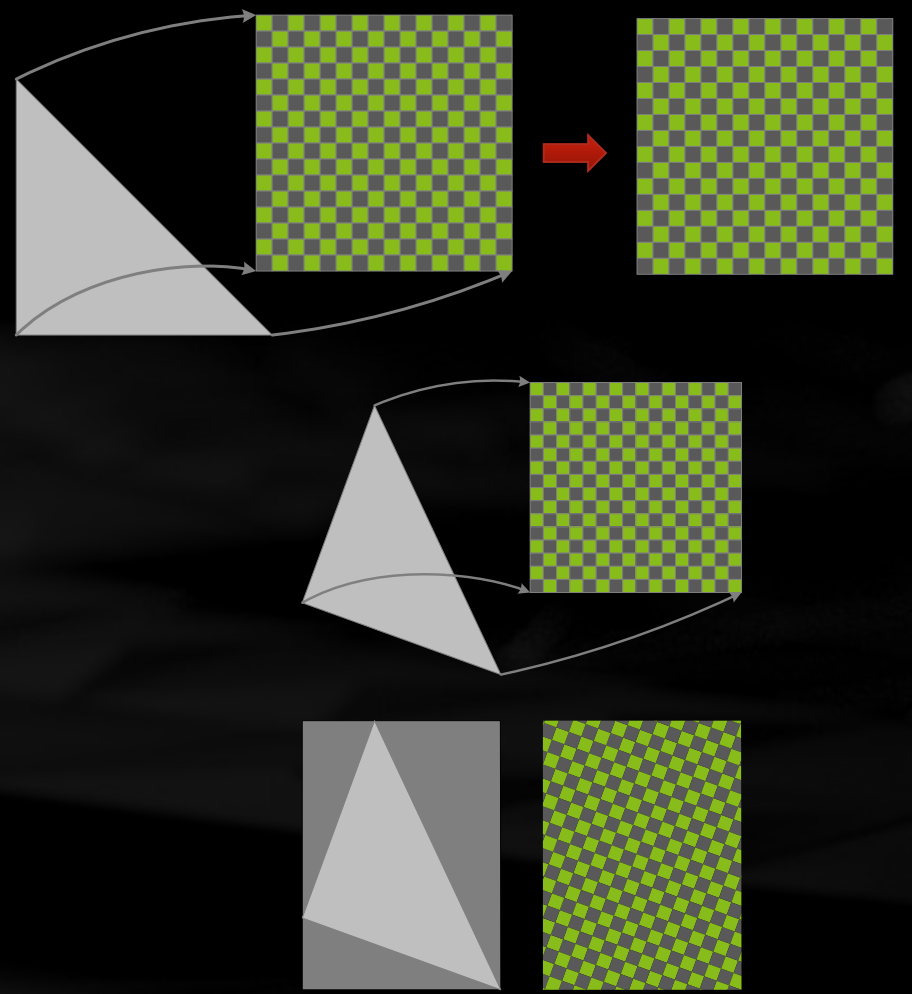
# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers



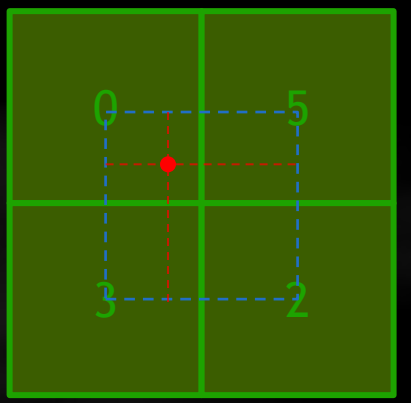
# Screen Space BBox Rasterization

- Screen Space Bbox rasterization
  - Reduce # of vertices  sent to GPU
  - Speeds up particle systems, point sprite etc.
- Supported by these APIs:
  - OpenGL:  NV_fill_rectangle extension
  - NvAPI:  coming soon

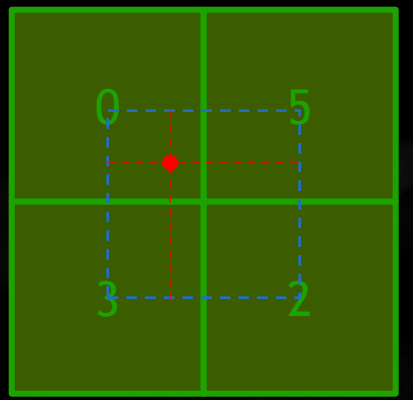


# Min/Max texture filtering

- Hardware support for min/max filtering
- Usecases:
  - Min-Max shadow maps
  - Other min-max reduction chains
- API support:
  - OpenGL: EXT_texture_filter_minmax
  - DirectX11.2



*MAX returns “5”*

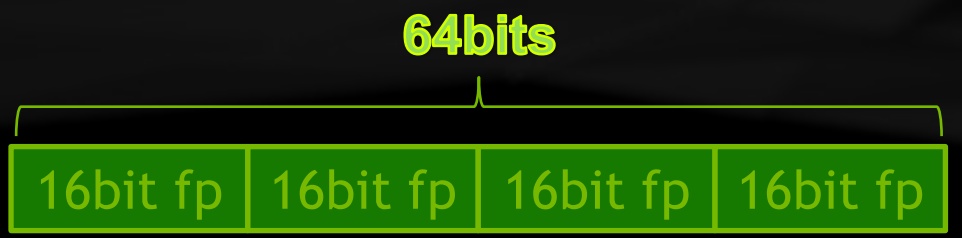


*MIN returns “0”*

# New Interlocked Operations 1

- 2D-vector: two 16bit floating point numbers


- 4D-vector: four 16bit floating point  numbers



# New Interlocked Operations 2

- Usecases:
  - Reduce the number of Interlocked ops during e.g. light accumulation
  - Save memory if you only need 16bit values
- API support
  - OpenGL + Android: NV_shader_atomic_fp16_vector
  - NvAPI: coming soon



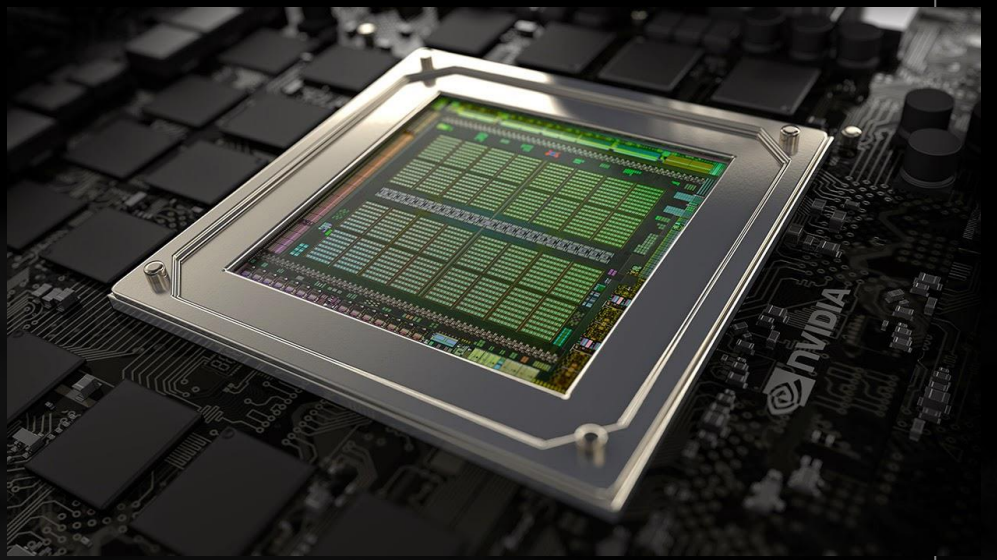
# Extended Blend Modes

- ZERO SRC
- DST
- SRC_OVER
- DST_OVER
- SRC_IN
- DST_IN
- SRC_OUT
- DST_OUT
- SRC_ATOP
- DST_ATOP
- XOR PLUS
- PLUS_CLAMPED
- PLUS_CLAMPED_ALPHA
- MULTIPLY
- SCREEN
- OVERLAY
- DARKEN
- LIGHTEN
- COLORDODGE
- COLORBURN
- HARDLIGHT
- SOFTLIGHT
- SOFTLIGHT_SVG
- DIFFERENCE
- MINUS
- MINUS_CLAMPED
- EXCLUSION
- CONTRAST
- INVERT INVERT_RGB
- INVERT_KHR
- LINEARDODGE
- LINEARBURN
- VIVIDLIGHT
- LINEARLIGHT
- PINLIGHT
- HARDMIX
- RED
- GREEN
- BLUE
- HSL_HUE
- HSL_SATURATION
- HSL_COLOR
- HSL_LUMINOSITY



# Outline of this talk

- Architectural goals of Maxwell
- DirectX12 hardware features
  - Conservative Rasterization
  - Raster Order Views
  - Tiled Resources
- Multi-Projection Acceleration
- New Antialiasing Features
- Misc other new features
- Questions and Answers



# GameWorks

- Get the latest information for developers from NVIDIA and continue the discussion
- gameworks.nvidia.com