CARBON

OptiX for DirectX Programmers: Raytracing Eve Online's Portraits

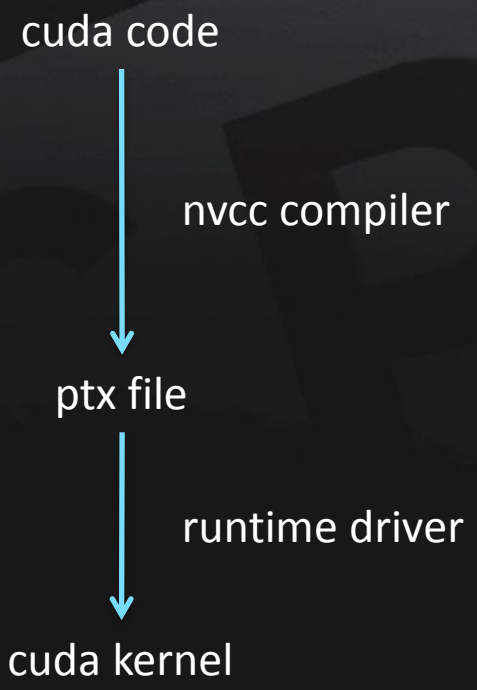
Bert Peers, CCP Games, GPU Technology Conference 2012 S0021 v7


Microsoft
DirectX
NVIDIA

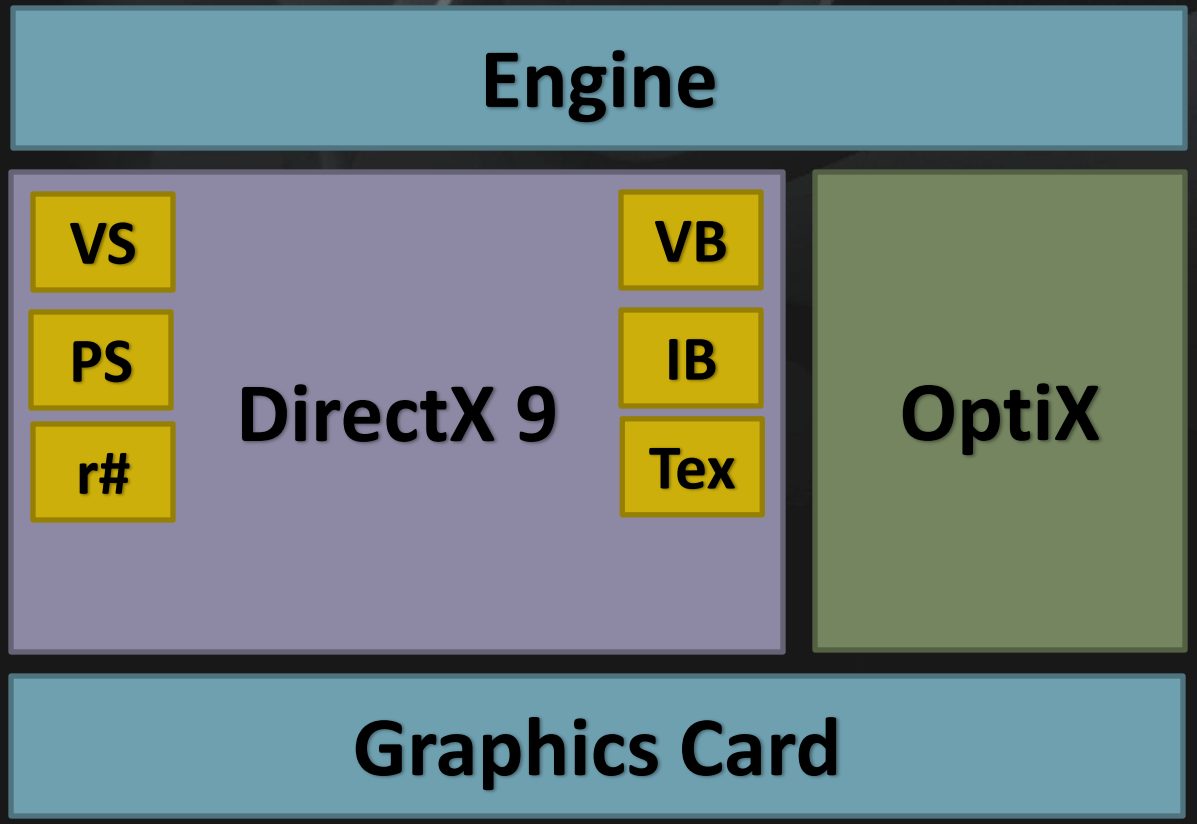
- I. OptiX and DirectX
- II. Your Data
  - What Works
  - What Doesn’t
  - Cheating
- III. Your Code
  - Translucency
  - Multipass

I. OptiX

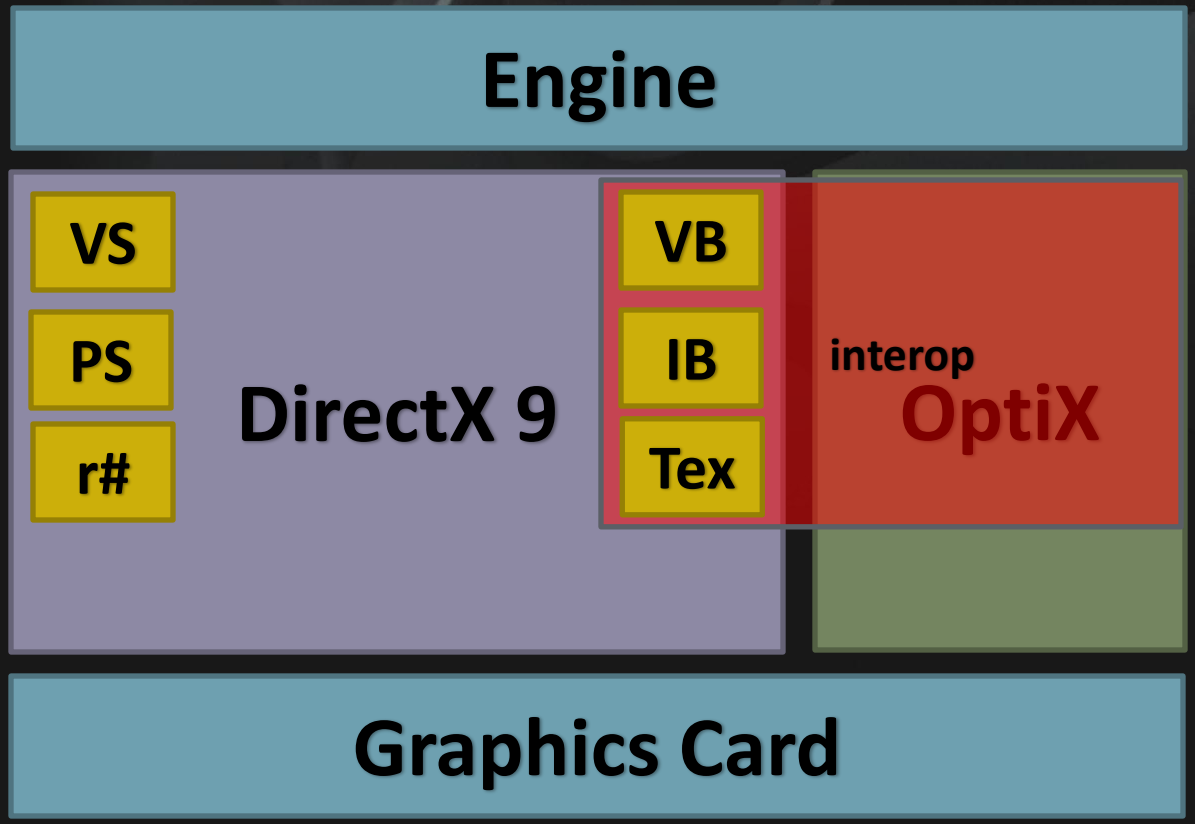
- GPU-based realtime raytracing
  - Including BVH setup
- Customizable black box
  - Intersection Tests
  - Computing AABB
  - Shading
  - (…)



I. Optix - InterOp



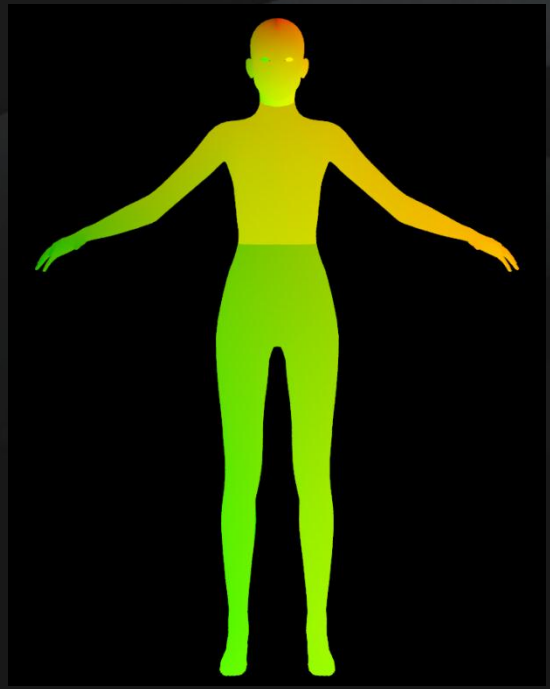
I. Optix - InterOp

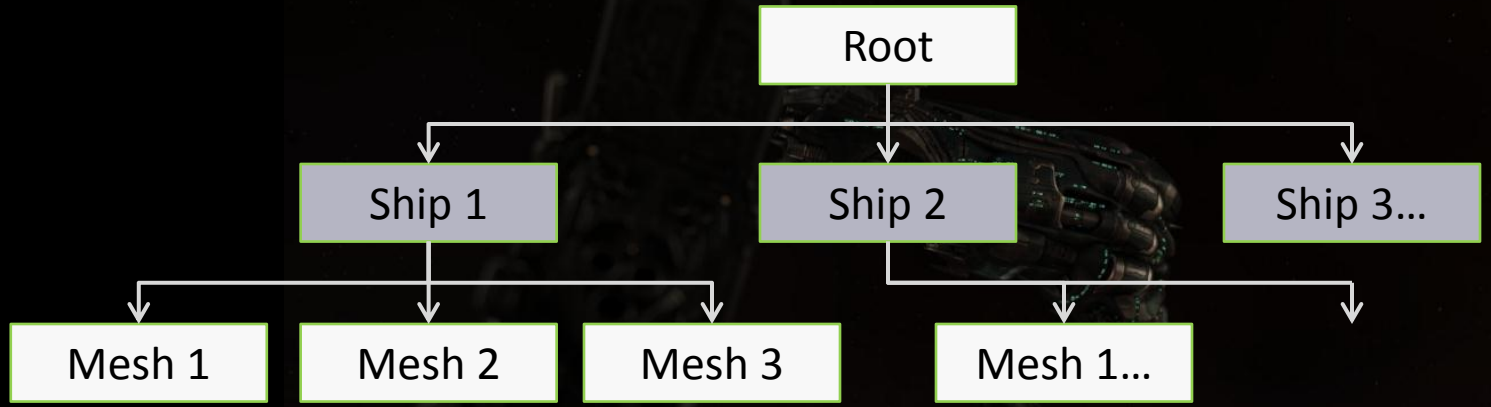


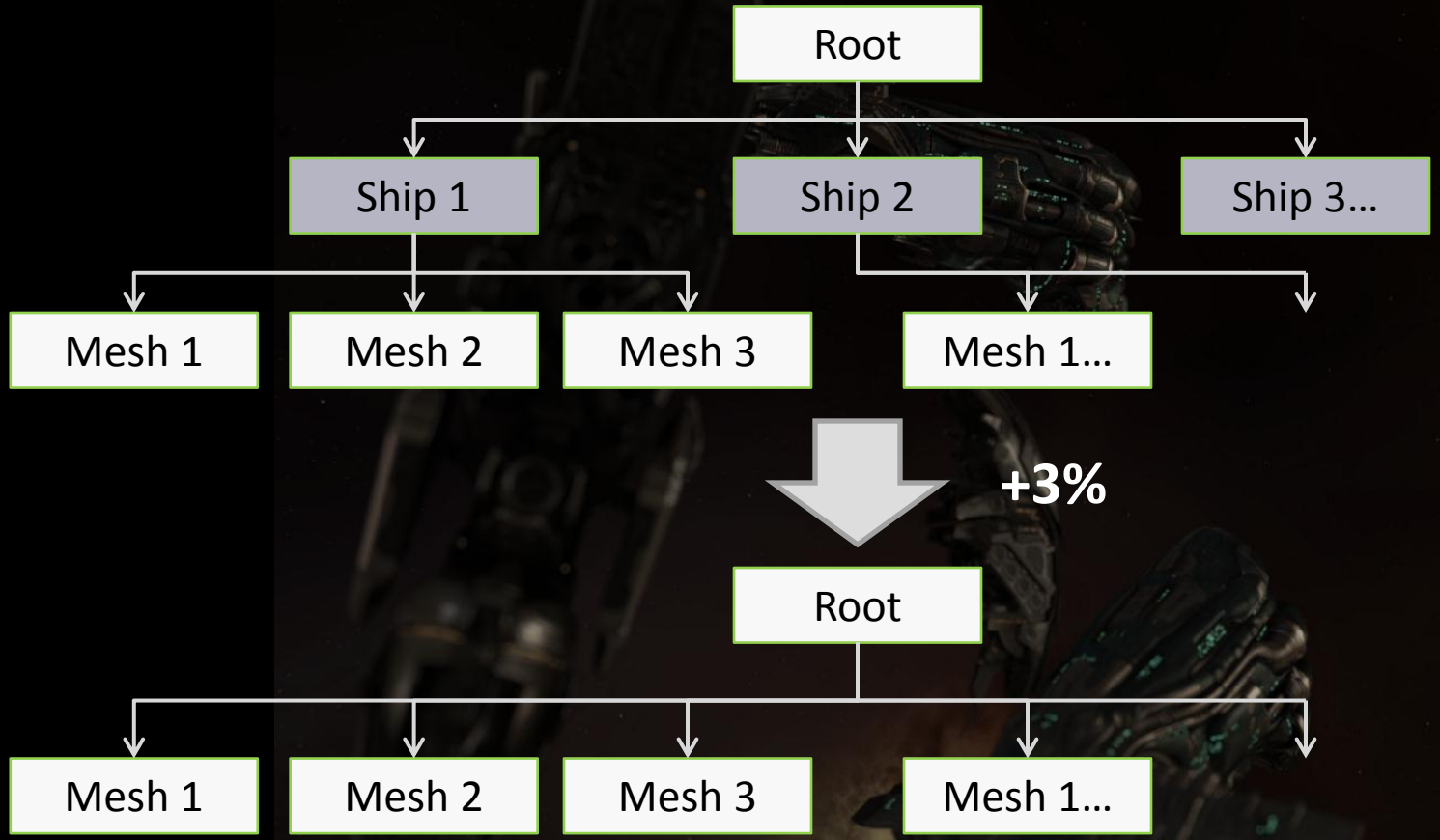
- I. OptiX and DirectX
- II. Your Data
  - Buffers
  - Skinning
  - Textures
  - Shaders
- III. Your Code
  - Translucency
  - Multipass

II.1. Vertex & Index Buffers

- Supported pool+flags
- Traverse scenegraph
  - Call `createBufferFromD3D9Resource`
  - Exactly once per VB or IB
- Exactly match memory layout
  - eg. `ushort + __half2float`
  - Use templates/ifdef
  - May impact BVH options

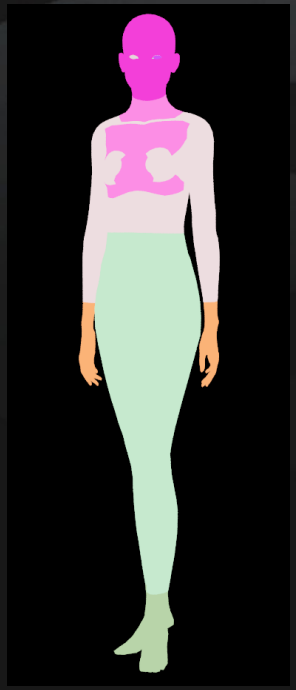



Root
Ship 1
Ship 2
Ship 3…
Mesh 1
Mesh 2
Mesh 3
Mesh 1…


Root
Ship 1
Ship 2
Ship 3…
Mesh 1
Mesh 2
Mesh 3
Mesh 1…
+3%
Root
Mesh 1
Mesh 2
Mesh 3
Mesh 1…

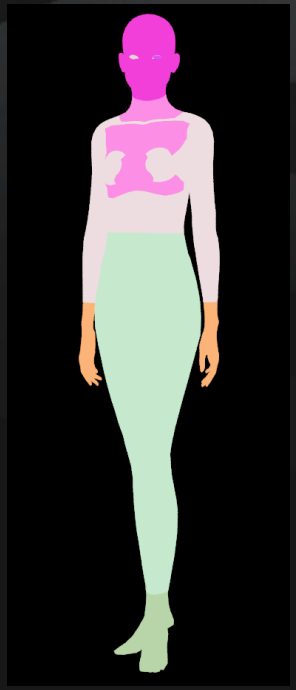
II.2. Skinning

- No Interop: new buffer, fill matrices
- When do you Skin?
- Option 1 : during intersection/AABB
  - Fetch vertex data
  - Skin positions
  - Test against skinned positions
  - Atrocious (-30%), but works



II.2. Skinning – Smart Way

- Better: skin once
- DirectX10+: use stream out
- Use OptiX itself
  - OptiX *is* Cuda!
  - rayGen without rtTrace() == Cuda kernel
  - Flexible: change format; PN-tessellate;…
  - Don’t need 3-way interop



II.3. Textures

- Supported pool + supported format + `createTextureSamplerFromD3D9Resource`
- Bad news
  - Short list of formats.. no DXT
  - No MipMaps
  - No CubeMaps

II.3. Textures

- Format: make supported RenderTarget
  - Full-screen-blit “bad” texture
  - Interop the RT instead
- Mips: see above.. Use SAT?
- CubeMaps: make 6 RTs, see above
  - Roll your own texCube… minus mips
- Remember to sRGB, swizzle

II.4. Shaders

- Fun Part: copy PS to Cuda, patch it up
- Copy register data to `rtDeclareVariable`
- Performance tips
  - Try to remove zombies
  - Try to tex2D up front
  - Exception Handler for NaN/inf

- I. OptiX and DirectX
- II. Your Data
  - Buffers
  - Skinning
  - Textures
  - Shaders
- III. Your Code
  - Translucency
  - Multipass

III.1. Translucency

- Need to blend in color of “behind me”
  - Shoot new ray, go get it
- Classic problems
  - Start too close: self intersect
  - Start too far: miss/tunnelling
- Use OptiX to…
  - … find all points with 1 ray
  - … avoid stop-restart problem

III.1. Translucency

- OptiX feature: any_hit
  - Called during closest hit discovery
- + `rtIgnoreIntersection`
  - = ray just keeps going
- Idea: ignore always & enumerate all hits
- Store them
- When ray is done, sort back to front, shade, blend

III.1. Translucency

- OptiX feature: any_hit
  - Called during closest hit discovery
- \+ `rtIgnoreIntersection`
  - = ray just keeps going
- Idea: ignore always & enumerate all hits
- Store them
- When ray is done, sort back to front, shade, blend

III.1. Translucency - Storage

- Bad: store hits in perRayData
  - Too big = bad performance, always
  - Too small = dropping random points
- Good: build linked list
  - prd.head = list root
  - Allocate nodes with atomicInc (sm20+)
  - Overhead only when used
  - Cfr. DX11 Order Independent Transparency [OIT10]

III.1. Translucency - Shading

- OptiX expects shading in closest_hit
- Per-material variables not visible in rayGen
  - But needed to shade
- “Solution”: dump a copy in list node
  - “Shader Closure” / Deferred Shading [Val07]
- Shade in `any_hit`? *Almost*… but No
- Wishlist: store link to Context?

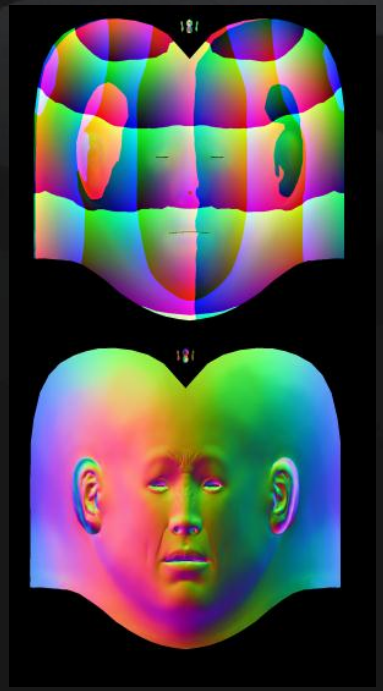
```
rtGetValue( someContext, “diffuseColor” )?
```

III.2. Multipass

- Typical: [d'Eon08]
  - Render to Texture
  - Filter the Texture
  - Render scene with Texture
- Easy port:
  - Raytrace to Texture
  - Filter the Texture
  - Raytrace scene with Texture

III.2. Multipass – A Better Way

- Idea: query data on demand
- Skip intermediate texture
  - cfr. Shadow ray versus shadow map
- Want averaged nearby diffuse?
  - Move around, shoot rays, filter results
  - = on the fly gather
  - Ideal with stochastic methods
  - UV worldpos maps to “move around”




off
on

Thank You!

- Special Thanks to Optix-Help & Austin Robison
- [OIT10] Real-Time Order Independent Transparency and Indirect Illumination using Direct3D 11, *Jason Yang, Jay McKee, Siggraph 2010*
- [d'Eon08] Advanced Techniques for Realistic Real-Time Skin Rendering, *Eugene d'Eon et al, GPU Gems 3*
- [Val07] Deferred Rendering in Killzone 2, *Michal Valient, Develop 2007*
- Making Faces: Eve Online's New Portrait Rendering, *Bert Peers, Siggraph 2011*

- Questions?
  - Nvidia Optix Forum