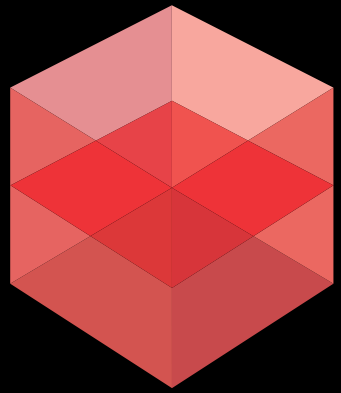


REDSHIFT

Production-quality, final-frame rendering on the GPU

Hello!

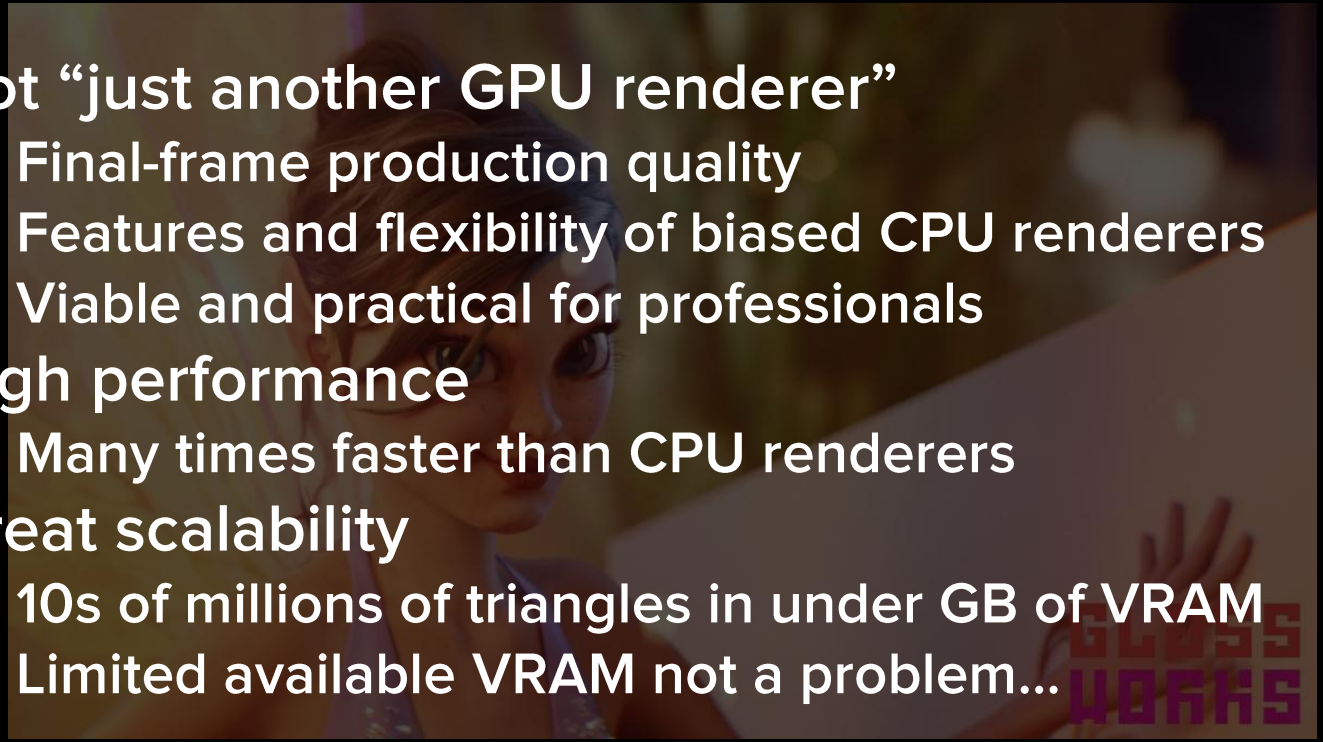
- Company founded in 2012
- Launched the first alpha in March 2013
- Version 1.0 in March 2014
- No marketing effort so far...

REDSHIFT

What is Redshift?

REDSHIFT

What is Redshift?



- Not “just another GPU renderer”
  - Final-frame production quality
  - Features and flexibility of biased CPU renderers
  - Viable and practical for professionals
- High performance
  - Many times faster than CPU renderers
- Great scalability
  - 10s of millions of triangles in under GB of VRAM
  - Limited available VRAM not a problem…

REDSHIFT

Breaking the Memory Barrier

- Virtually unlimited number and size of bitmap textures
- Out-of-core paging technology
- Built in UDIM/UVTILE texture tile support

REDSHIFT

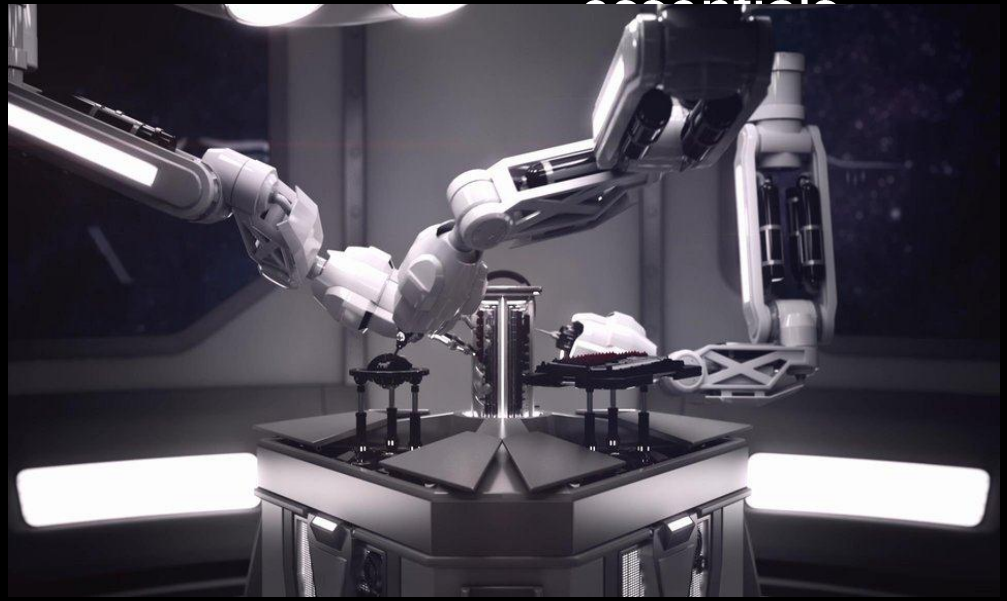
Some Essentials

- Flexible material system
- Multiple ‘biased’ accelerated GI modes
- Full samples control for noise clean-up

REDSHIFT

More Cool Features

- Efficient volumetric scattering
- Vector displacement mapping
- Object visibility flags
- Light/Shadow linking
- Compositing
- And many more essentials



REDSHIFT


BRAD PITT
WAR NEVER ENDS QUIETLY
FURY
NOVEMBER
Join us at r/Fury


MAGNETIC | DREAMS
Chick-fil-A


STALLONE STATHAM BANDERAS LI SNIPES LUNDGREN GRAMMER COUTURE CREWS LUTZ
ROUSEY POWELL ORTIZ DAVI GIBSON FORD SCHWARZENEGGER
NEW TEAM, NEW ATTITUDE, NEW MISSION
THE EXPENDABLES III
THURS AUG 14
#EX3


STAFF PICK
WINNER BEST SHORT FILM LAS VEGAS FILM FESTIVAL 2014
WINNER BEST ANIMATION CATALINA FILM FESTIVAL 2014
WINNER BEST ANIMATION ASHEVILLE CINEMA FESTIVAL 2014
WINNER HONORABLE MENTION LA UNDERGROUND FILM FESTIVAL 2014
DANNY and The WiLD BUNCH
OFFICIAL SELECTION BEVERLY HILLS SHORTS FESTIVAL 2015
OFFICIAL SELECTION SHORTS BACKLOT FILM FESTIVAL - SPAIN 2015
OFFICIAL SELECTION ATLANTA SHORTS FEST 2014
OFFICIAL SELECTION CUCALORUS FILM FESTIVAL 2014
OFFICIAL SELECTION CINEQUEST FILM FESTIVAL 2015
OFFICIAL SELECTION SEDONA INTERNATIONAL FILM FESTIVAL 2015
OFFICIAL SELECTION LAS VEGAS FILM FESTIVAL 2014
OFFICIAL SELECTION GOLDEN DOOR FILM FESTIVAL 2014
OFFICIAL SELECTION BAHAMAS INTERNATIONAL FILM FESTIVAL 2014
OFFICIAL SELECTION CLEVELAND INTERNATIONAL FILM FESTIVAL 2015
OFFICIAL SELECTION IMAGINARIA ANIMATION FILM FESTIVAL - ITALY 2015

REDSHIFT


GLASS
WORKS

Challenges : Out Of Core (1)

- Allows scene data size exceed the GPU mem limits
  - Geometry and textures
- GPU geometry/texture caches
- What about direct CPU memory reads?
  - Certain limitations prevented that
  - Unified virtual addressing (UVA) needs pinned memory
  - Unified memory doesn’t work like a cache
- In the end, we designed our own caching system

Challenges : Out Of Core (2)

- Communication between the GPU and the CPU
  - Stalls
  - WDDM overheads
- Blocking / granularity of data is important!
  - Tiled mipmaps for textures
  - Local primitive groups for geo
- The GPU caches work *really* well with textures
  - Great cache-hit ratios for tiled mipmaps!
- Geometry can cause performance issues, though
  - But we have some interesting ideas about that
  - GTC2016 talk perhaps? ☺

REDSHIFT

Challenges : Material/Shader System

- More than 200 different shading nodes!
  - *Many* combination possibilities!
  - Hardcoded system would be insufficient
- Original solution: recompile on shader graph edit
- In the end: function pointers
  - Each node is a function
  - No slow recompilation needed
  - Contemplating CUDA 7 runtime compilation
- Efficient material blending/layering
  - Incorporated importance sampling

REDSHIFT

Challenges : Multiple Kernels (1)

- Two schools of thought:
  - Uber-kernel (single kernel launch) vs multiple kernel launches
  - Different pros and cons
- Had to go with multiple kernels because:
  - Out of core data access
  - Multiple GI modes
  - Shading
- Reasons we didn’t go with uber-kernels:
  - Compiler issues (bugs, long compilation times)
  - Register pressure (also a compiler issue)
  - Hard to debug / profile

REDSHIFT

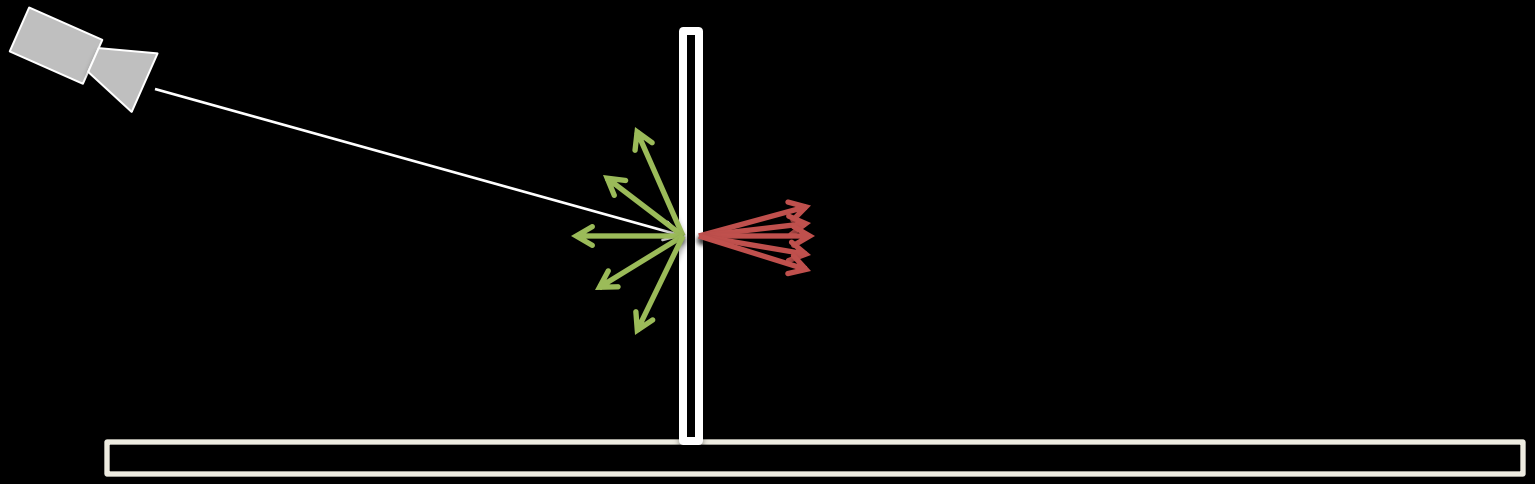
Challenges : Multiple Kernels (2)

- Splitting into multiple kernels benefits
  - Helps with divergent branches:
    - Sorting
    - Rescheduling / Persistent threads
    - Easier to do with multiple kernels than uber-kernels!
  - Helps rapid prototyping of new code
- Drawbacks
  - Overheads
  - GPU underutilization
  - Have to keep the GPU busy
    - Group as much work as possible

REDSHIFT

Challenges : Ray Generation (1)

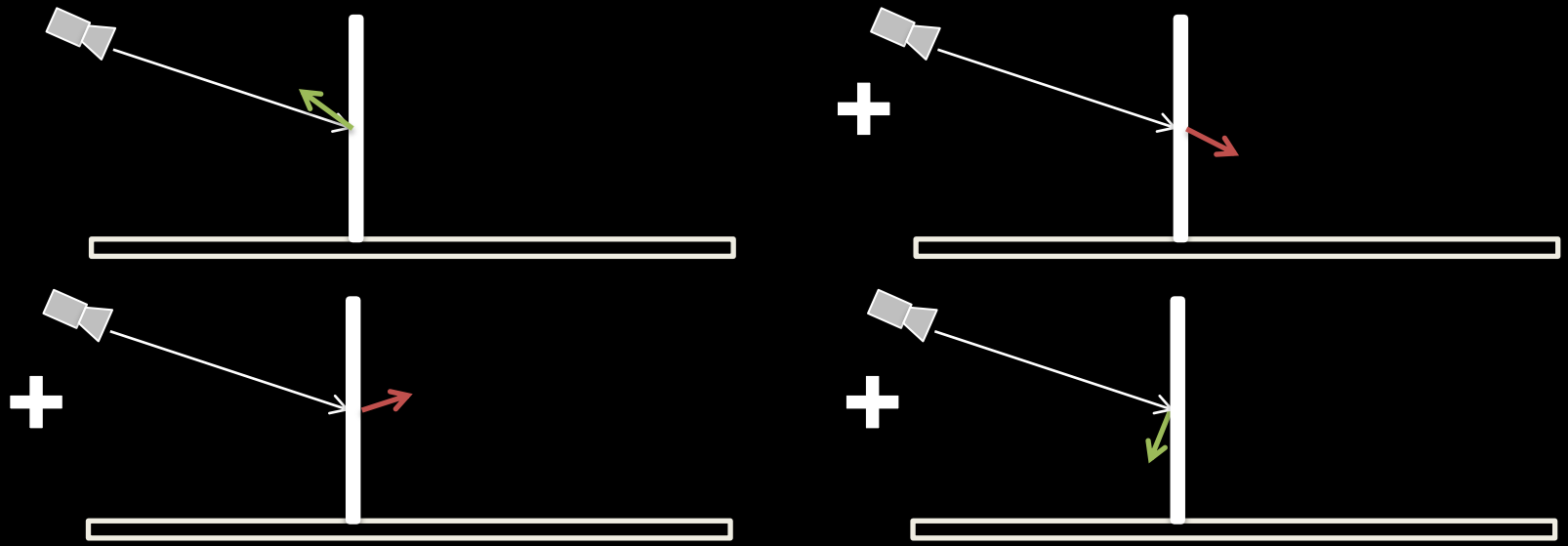
- The life of a ray (glossy reflective/refractive plane)
- So one camera ray = multiple reflection/refraction rays



REDSHIFT

Challenges : Ray Generation (2)

- Alternative: Russian Roulette
  - Instead of executing all possibilities, it picks fewer (or just one)
  - Renders in “passes”



Challenges : Ray Generation (3)

- Here we showed glossy reflection/refraction
  - Can also be used for GI, direct lighting, etc
- Great for interactive (progressive) rendering!
- Picking “the right rays” at each pass is hard!
  - Several rendering techniques for this
  - If the technique doesn’t do a good job → noise!
  - …and having to start a new ray from the camera
- Russian roulette-style rendering is easy for artists
  - Just a single “number of passes” setting
  - No sample tweaking necessary on a per light or surface basis

REDSHIFT

Challenges : Ray Generation (4)

- Russian roulette simplifies several things in a renderer
  - Popular choice among current GPU renderers
- Redshift works a bit differently
  - Each ray can spawn multiple rays, when necessary
  - This is what most “biased” CPU renderers do
  - Certain scenes clean up faster and with fewer rays
  - It gives artists more control over noise
- But Redshift can also do russian roulette!
  - Used for progressive rendering
  - Used depending on ray importance

REDSHIFT

Challenges : Ray Generation (5)

- Spawning multiple rays from a single ray is tricky!
  - Trivial solution: spawn rays within a loop
    - Hard to optimize / complicated kernel
    - Can destroy coherency / performance
    - There exist solutions, but are a bit messy
  - Better: spawn rays that are processed in parallel!
    - Job Queue
    - Useful for material layering, multiple lights, etc
  - Downside: Harder to manage memory
  - Worth it!

REDSHIFT

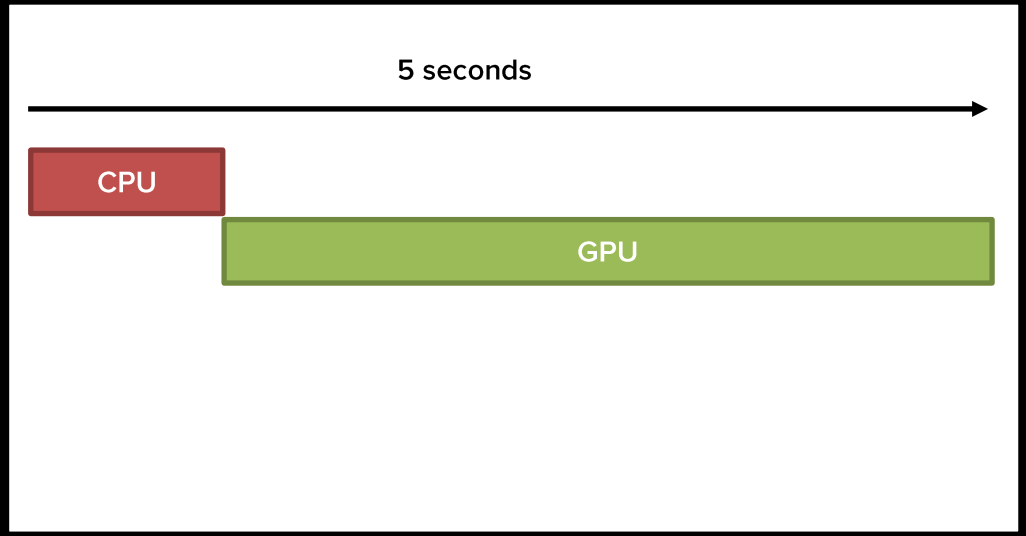
Challenges : CPU Performance (1)

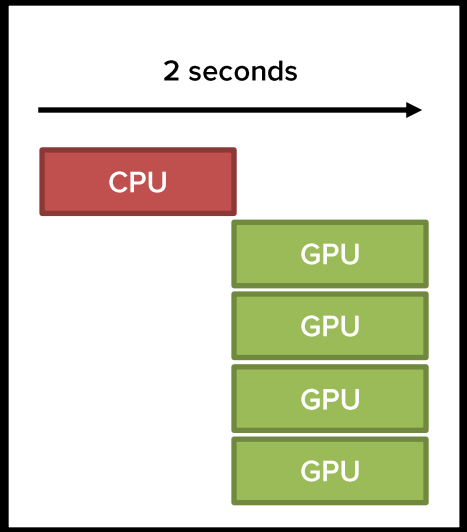
- GPU can’t (currently) do everything by itself!
  - Scene extraction from the 3D app
    - Often times, single threaded!
  - Disk/Network data loading
    - Meshes / textures / etc
  - Construction of ray tracing acceleration structures
  - Tessellation
  - Certain parts of our out-of-core algorithms
  - Saving final images to disk
    - DeepEXR saving can be slow!

REDSHIFT

Challenges : CPU Performance (2)

- The faster the GPU rendering, the more the CPU performance matters!





REDSHIFT

Challenges : CPU Performance (3)

- Solutions
  - Speed up CPU processing (duh!)
    - Multithread/optimize as much as possible!
    - Cache/reuse data
    - Involve the GPU more during preprocessing
    - …an ongoing effort for Redshift!
  - Render a few frames at once
    - Forces “multithreading” on CPU processing
    - …but consumes more main memory

REDSHIFT

The Future

- 3ds Max plugin currently in alpha!
- Support for more DCCs (Cinema4D, Houdini, Modo, Blender)
- Linux support (in progress)
- Ray marching (non-homogeneous volumetrics)
- Single scattering
- Shave/Yeti/XGen support in Maya
- More BRDFs (GGX)
- User AOVs (regular AOVs already supported)
- User defined shader nodes (shader SDK)
- More optimizations!

REDSHIFT

Thanks!

- For more information, please contact us at info@redshift3d.com
- Or meet us
  - …right after this presentation!
  - …at tonight's welcome reception
  - …at our demo area in the GTC Expo hall:
  - …Booth #112

REDSHIFT