When will Ray-Tracing Replace Rasterization?

Philipp Slusallek

Computer Graphics Lab
Saarland University, Germany
http://graphics.cs.uni-sb.de

Short Answer: The Process Started Last Year

- **The technology is available now**
  - Interactive RT on commodity HW (Cluster & GPU)
  - Support for massive and dynamic scenes
  - Support programmable & asynchronous shading
  - API for ray tracing (OpenRT)
- **Industry is highly interested**
  - E.g. Intel, AMD, Airbus, BMW, DC, VW, Audi, …
  - Solves many problems with rasterization
  - Greatly simplifies rendering

What is Possible Today?

- **OpenRT Performance**
  - Highly optimized software implementation
  - Well over 500 kRays / CPU
    - OpenGL shading: 2 fps @ video resolution
  - Interactive global illumination
    - Fully recomputed up to 5x per second
  - Logarithmic dependence on scene size
    - Interactively renders massive scenes (> 10 Mtris)
  - Linear scalability in number of rays and CPUs
    - Cluster computing for shadows, reflection, …

What is Possible Today?

- **Hardware Configuration for Videos**
  - Eight dual AMD Athlon 2000+ PCs
  - Fast-Ethernet connection to switch
  - Gigabit-Ethernet from switch to master & display

What is Possible Today?

- **Videos (~5 min)**
  - PowerPlant: massive model, OpenGL, dynamics
  - Terrain: massive, OpenGL, exact shadows, dynamics
  - Office-shading: progr. shaders, PnP, no approx.
  - Museum: freeform dynamics
  - BMW: shading, curved reflectors, HDR
  - Headlights: complex effects, curved reflectors, HDR
  - InvDate: Effects of GI, smooth illum, interactive
  - Conference: subtle lighting, convergence, per pixel

Advantages over Rasterization

- **More Efficient**
  - Output sensitive / demand driven computation
  - Logarithmic scaling with scene size
  - Requires less memory bandwidth
  - Asynchronous shading
  - Less pre-computation and textures access
  - Easily scalable to multiple processors

Advantages over Rasterization

- **More Flexible**
  - Efficiently computes single rays (vs. regular grids)
  - Basis for many applications (physics, lighting, …)
- **Simpler Programming**
  - No restriction due to pipeline model
  - No approximations and obscure parameters
  - Direct use of high level shading language
  - Shaders are independent & combine freely
  - Global Illumination is easily supported

SaarCOR Ray Tracing Hardware

- **Features**
  - Scalable hardware design
    - Choose # of Traversal, Intersection & Shading units
  - Very small memory bandwidth
    - Most scenes require only standard 133 MHz SDRAM
  - Resources & performance
    - Half the floating point units as GeForce3
    - Roughly same performance (Quake)
  - Supports all ray tracing options

Outlook

- **RT will quickly become mainstream**
  - "It is the right way to do 3D graphics"
- **RT in HW**
  - Streaming Processor (GPU)
  - Extended CPUs
  - Special purpose HW (for games?)
- **API support: OpenRT**
  - Crucial for application support

Ray Tracing Events at Siggraph

- **Showfloor**
  - inTrace: RT on PC cluster (AMD booth)
  - Utah: 128 CPU Onyx (SGI booth)
  - Stanford HW: Nvidia booth (??)
- **Papers:**
  - Purcell, RT on Programmable Graphics Hardware, Fri, 11am
- **Sketches:**
  - How CSG-Based RT Saves Time …, Wed, 2:30pm
- **Panel**
  - DemoScene Panel, Wed, 3:30pm

Links

- **inTrace: Realtime RayTracing Technology**
  - http://www.intrace.com
  - Or see us at AMD booth
- **OpenRT:**
  - http://www.openrt.de
- **Graphics Lab, Saarland University**
  - http://graphics.cs.uni-sb.de