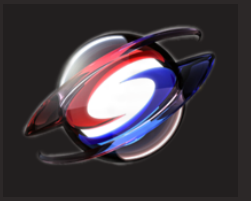
Beyond Programmable Shading Course
ACM SIGGRAPH 2012

Panel: From Publication to Product: How Recent Graphics Research has (and has not) Shaped the Industry

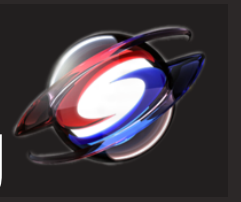
BILL MARK (START-UP COMPANY)

My Experience:
Two cycles of academia -> industry



- GPU Real-time programmable shading
  - Stanford & UNC -> NVIDIA
- Real-time ray tracing
  - Univ. of Texas at Austin & Stanford -> Intel

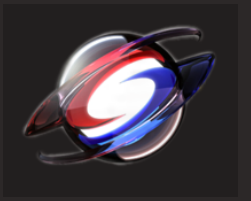
History of GPU Real-time programmable shading



- Academia: Stanford Real Time Shading Language
  - High level language & compiler
  - Clean abstractions

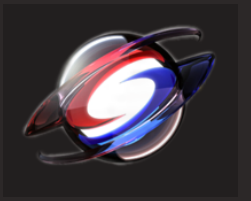
- Industry:
  - NVIDIA Cg
  - Microsoft HLSL
  - OpenGL GLSL

Primary impact was on language



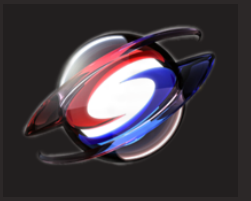
- Industry already planning programmable HW
- But, interface was lower level
  - Focus on assembly-level interfaces
  - HLSL initially vertex-only, not fragment/pixel
- Academic impact:
  - Accelerate high-level language, esp. for fragment/pixel
  - Better abstractions in industry systems
  - Enable and accelerate GPGPU

History of Real-time ray tracing



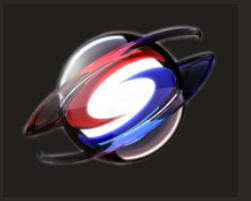
- Academia:
  - Fast traversal algorithms
  - Dynamic scenes
  - Systems (esp. Utah RTSL)
- Industry:
  - NVIDIA OptiX
  - Intel Embree
  - Autodesk/OptiCore, RTT, etc.

Impact different than anticipated



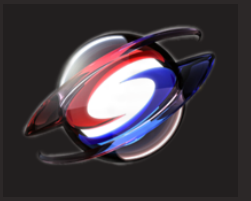
- Minimal use of ray tracing in games (so far)
  - Adoption likely to be incremental
  - Value shift: Emphasis shifting to device portability
- Significant impact on professional graphics
  - Autodesk, RTT, Mental Images, …
  - Better economic argument
    - Games: create once, view millions of times
    - Professional graphics: create once, view once

Recommendations for academia



- Don’t compete with industry
  - Lead in time, and/or
  - Cooperate & complement
- Focus on fundamental ideas & system abstractions
  - Prove new ideas and reduce risk

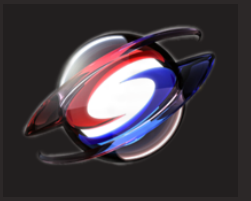
Future graphics systems research



- General strategy
  - Develop a vision of how things will be in 5-10 years
  - Identify problems that need to be solved
  - Solve problems and build prototype systems

- Specific topic: What does client+cloud system look like?
  - How is data and compute partitioned?
  - Most open area is image/video input and analysis
  - Metrics: ! / Watt and ! / wireless_byte
  - Leverage cloud benefits: multi-user & infinite data storage

Acknowledgements - collaborators



- Programmable shading
  - Academia: Kekoa Proudfoot, Pat Hanrahan, Tim Purcell, Marc Olano, …
  - Industry: NVIDIA team – Steve Glanville, Kurt Akeley, Mark Kilgard, …
- Real-time ray tracing
  - Academia: Warren Hunt, Peter Djeu, Gordon Stoll @ Intel, Don Fussell, …
  - Industry: Intel team – Sven Woop, Manfred Ernst, Ingo Wald, Carsten Benthin, …

Beyond Programmable Shading Course
ACM SIGGRAPH 2012