SIGGRAPHASIA2008

NEW HORIZONS

Parallel Computing for Graphics: Beyond Programmable Shading

SIGGRAPHASIA2008
NEW HORIZONS

Parallel Computing for Graphics: Beyond Programmable Shading

Introduction

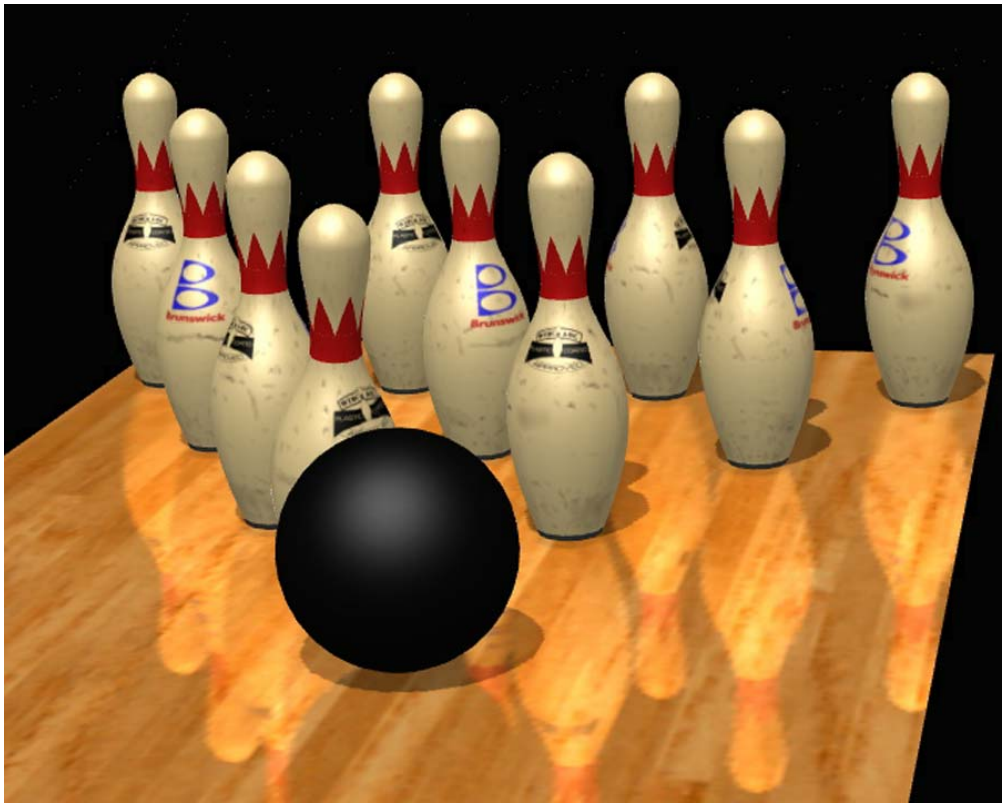
Anselmo Lastra

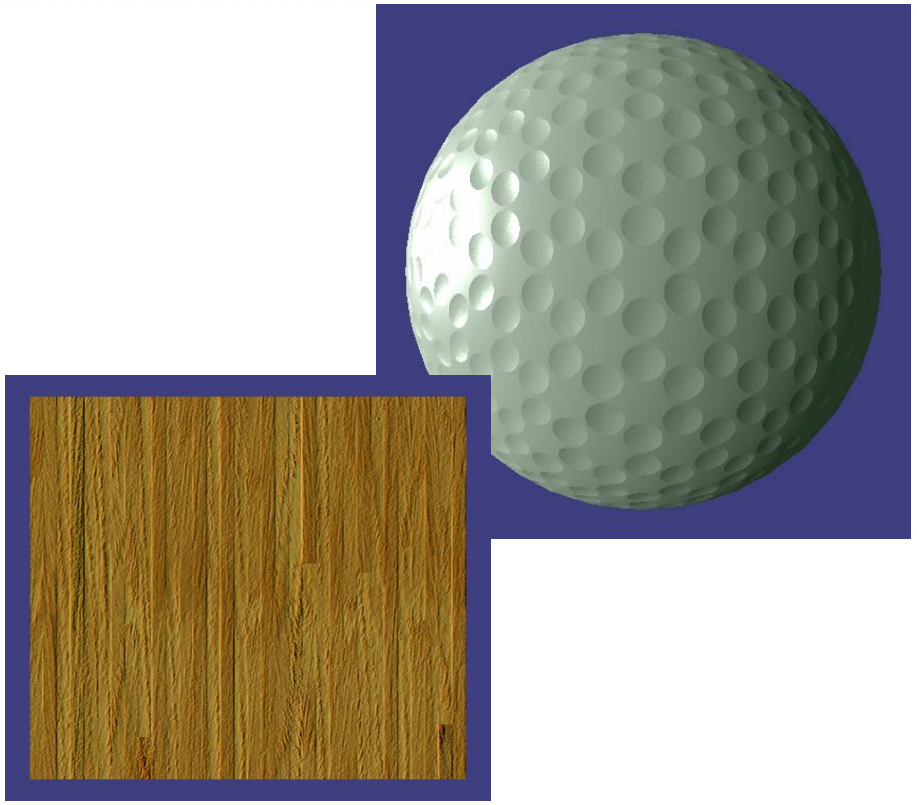
University of North Carolina at Chapel Hill

SIGGRAPHASIA2008

NEW HORIZONS

State of the Art, 1998





Marc Olano, Anselmo Lastra: A Shading Language on Graphics Hardware: The Pixelflow Shading System. SIGGRAPH 1998

Short Video

SIGGRAPHASIA2008
NEW HORIZONS

Enablers for Throughput Computing

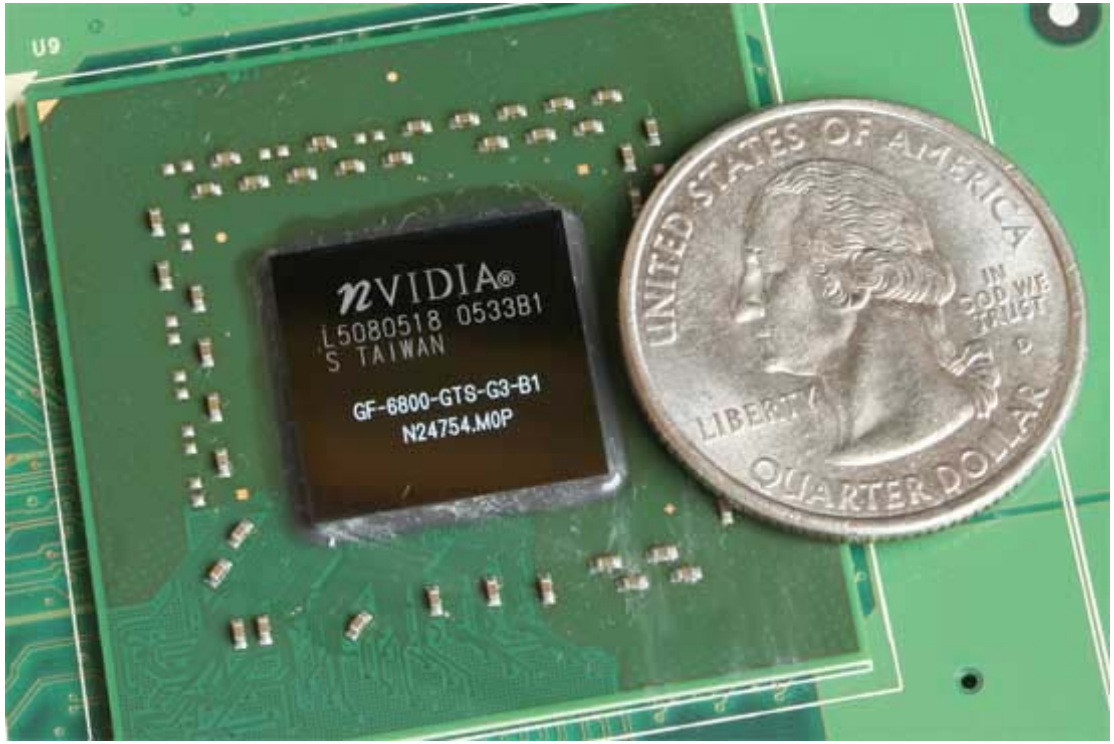
- ✓ High level language programming

Unfortunately, not Cost Effective!

- This was a prototype
- Meant to test concept
- Hundreds of chips

The Single Chip Revolution

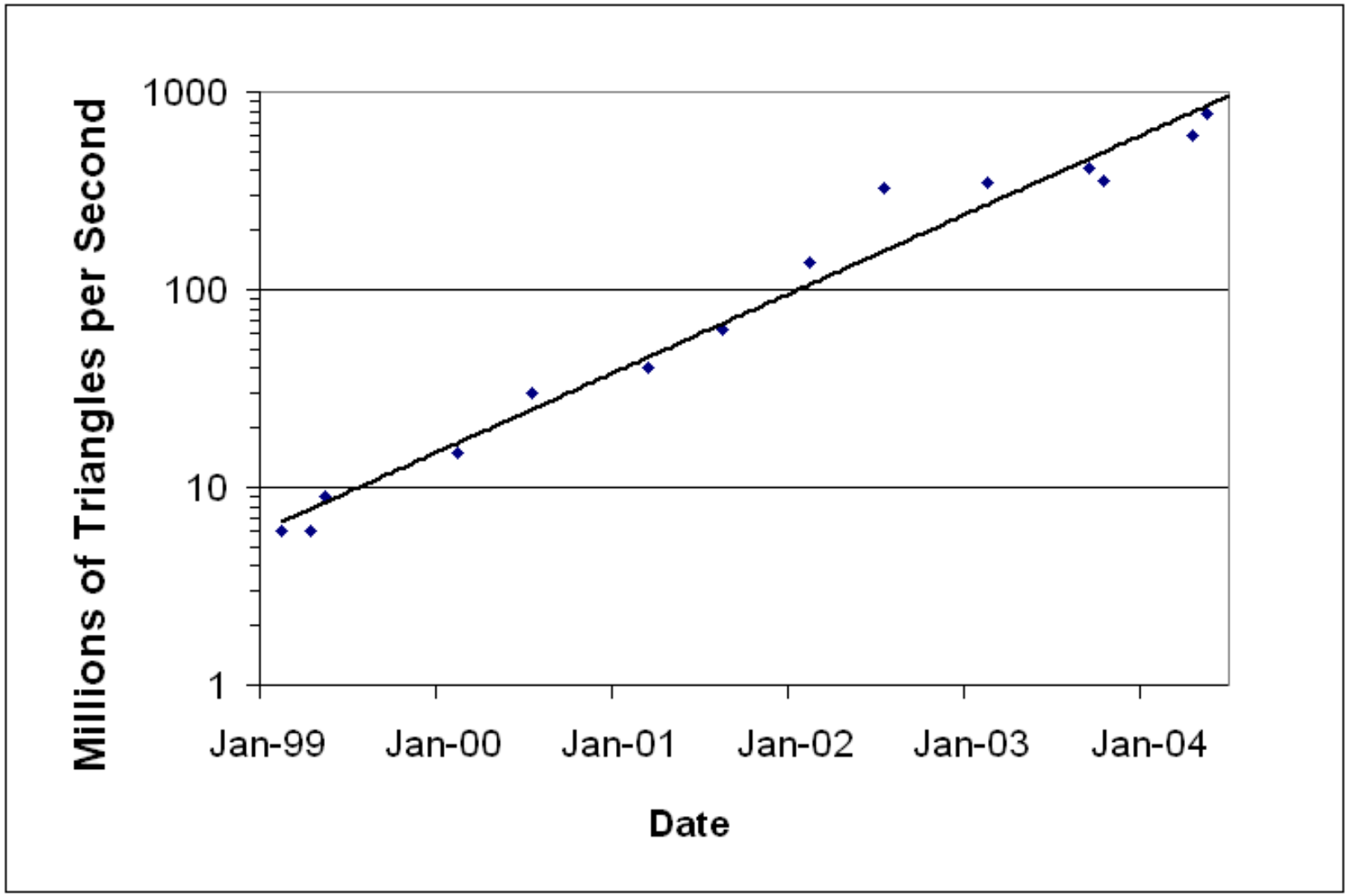
- As in CPUs, going to a single chip made for cost efficiency (as did the fact that PC graphics is a volume market)



Enablers for Throughput Computing

- ✓ High level language programming
- ✓ Excellent cost performance

Skyrocketing Performance



Why Such a Fast Rate of Increase?

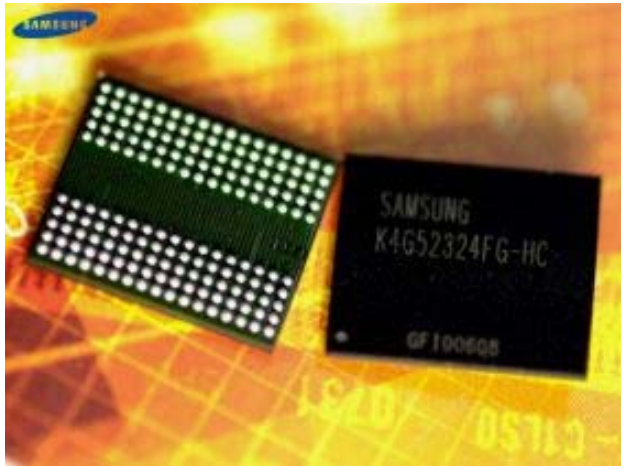
- Architectures meant for throughput, not for single thread
  - Not complexity of out-of-order issue
  - Nor of multiple issue per core
- Rendering is latency tolerant
- Graphics generates many parallelizable tasks
- Therefore much easier to scale

Enablers for Throughput Computing

- ✓ High level language programming
- ✓ Excellent cost performance
- ✓ Rapid increases in performance over time

Memory Bandwidth

- GPUs have driven high-performance memory market
- The GDDR series of DRAMs



Enablers for Throughput Computing

- ✓ High level language programming
- ✓ Excellent cost performance
- ✓ Rapid increases in performance over time
- ✓ High memory bandwidth

What's Left?

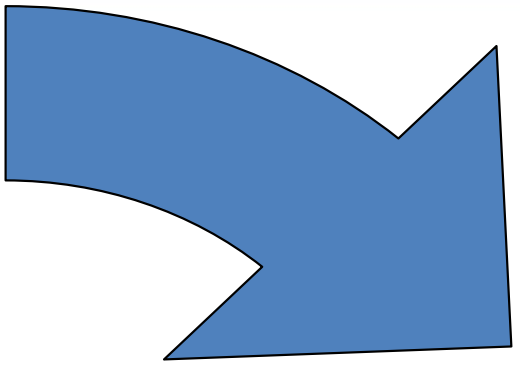
- Major stumbling block: special purpose nature of graphics hardware

- Let's look at the Wheel of Reincarnation

Wheel of Reincarnation

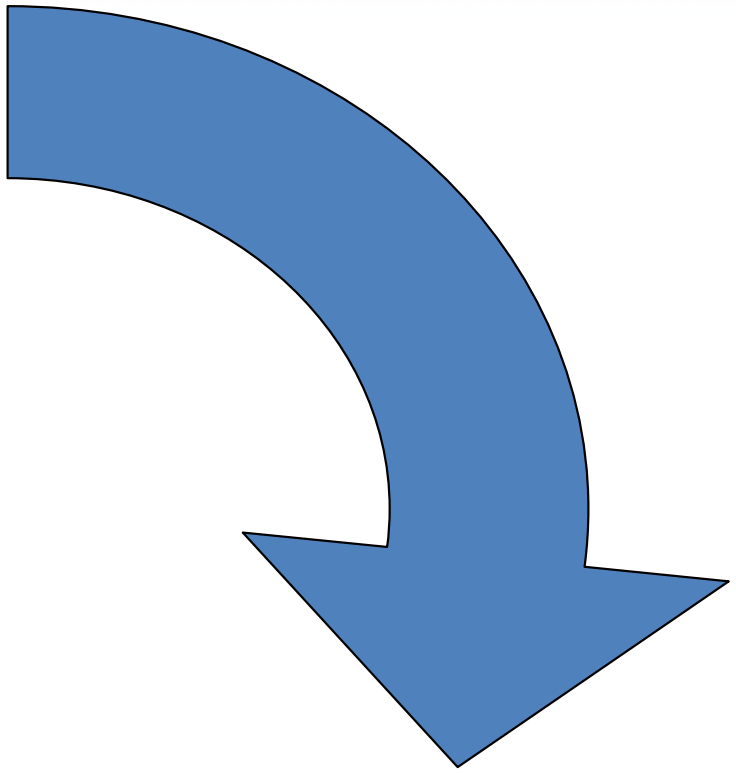
- CACM paper by Myer and Sutherland in 1968
- Tongue in cheek look at design of a display processor
  - In those days it was for vector graphics
- Authors examine their design process
- Let’s look and see whether it’s familiar

A Spin Around the Wheel



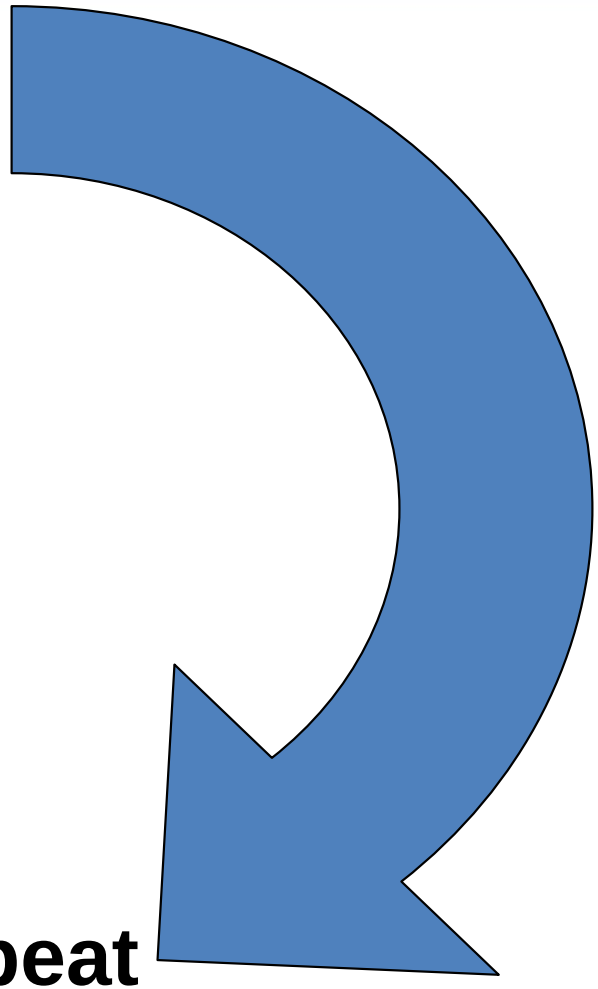
Display Channel for Points

A Spin Around the Wheel



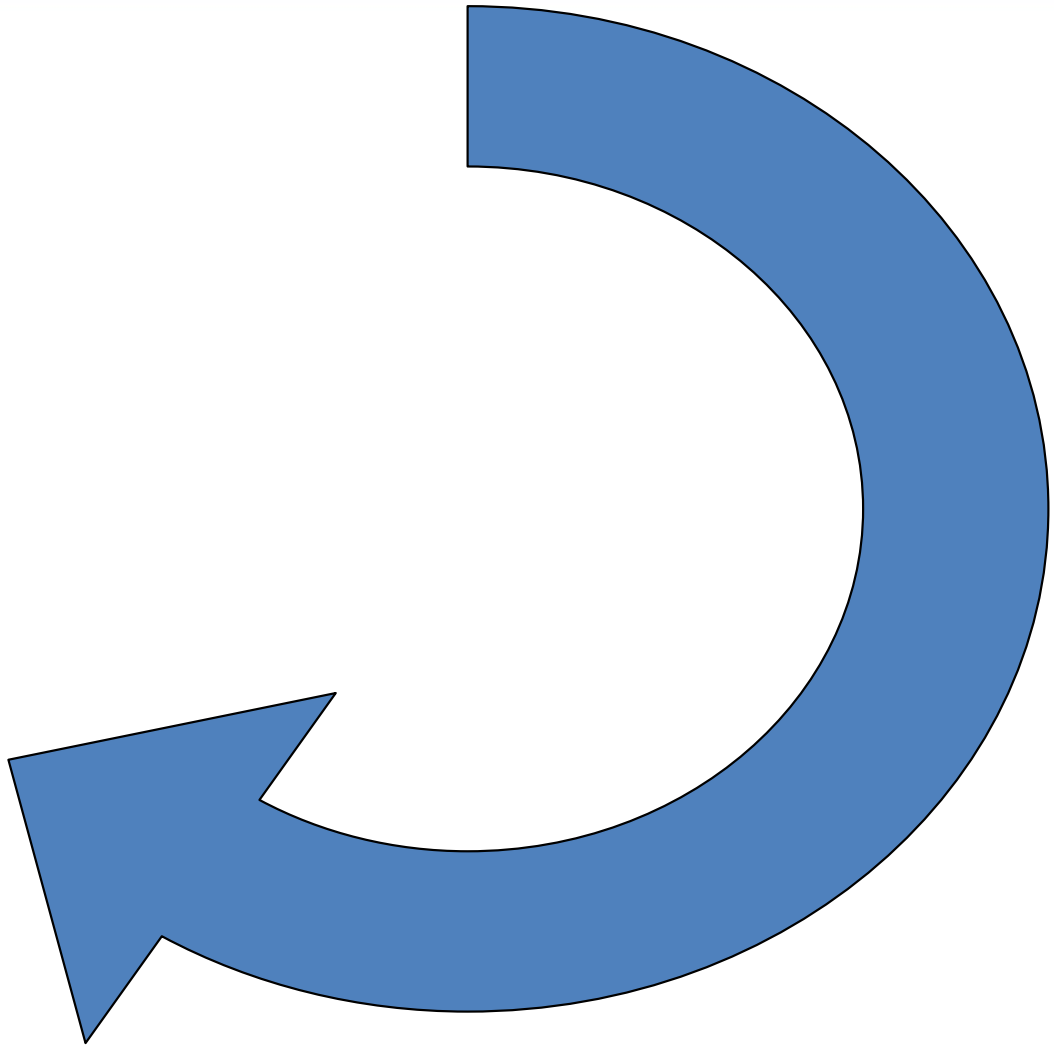
Line drawing hardware

A Spin Around the Wheel



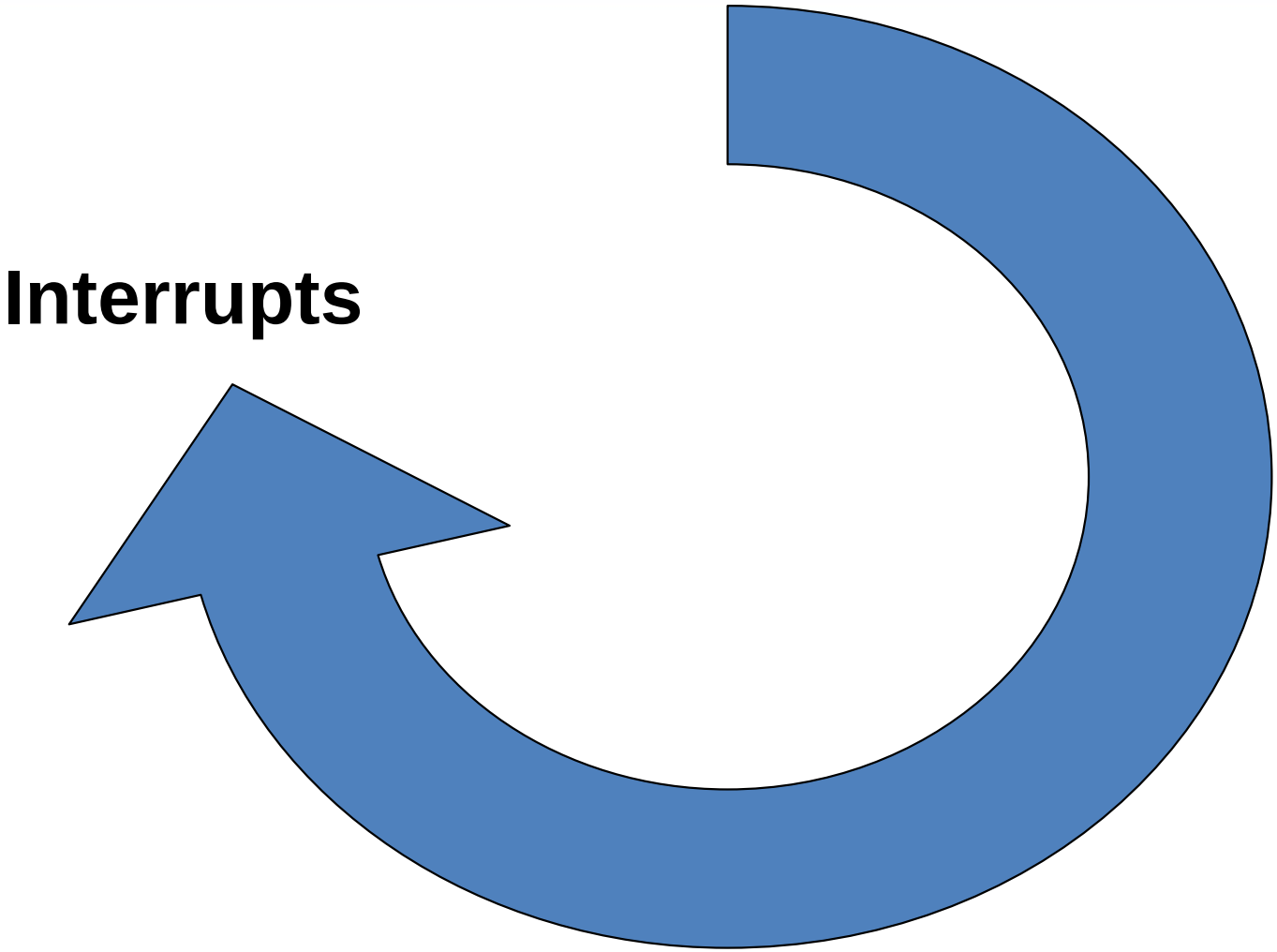
Jump to repeat

A Spin Around the Wheel

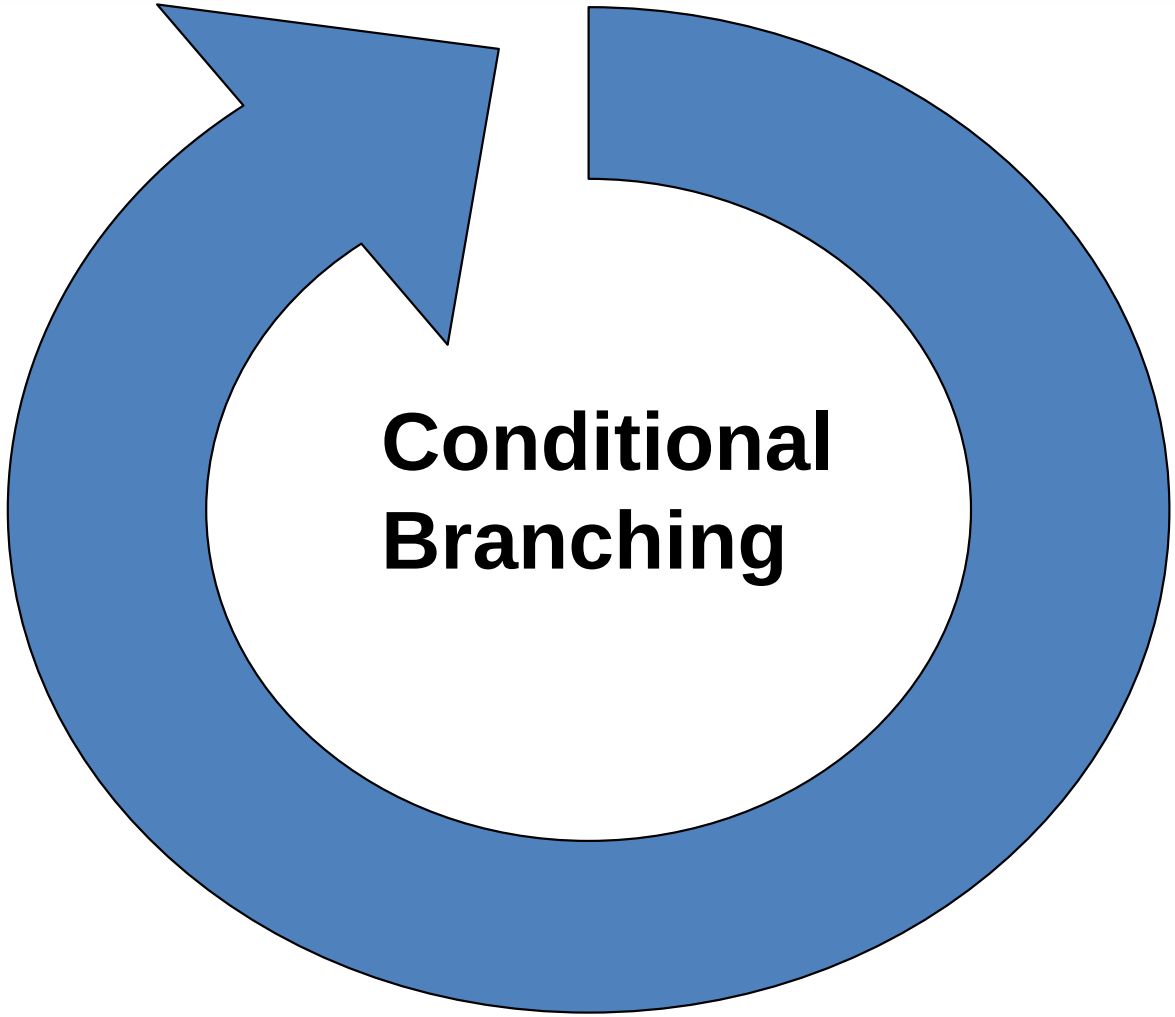


Subroutines

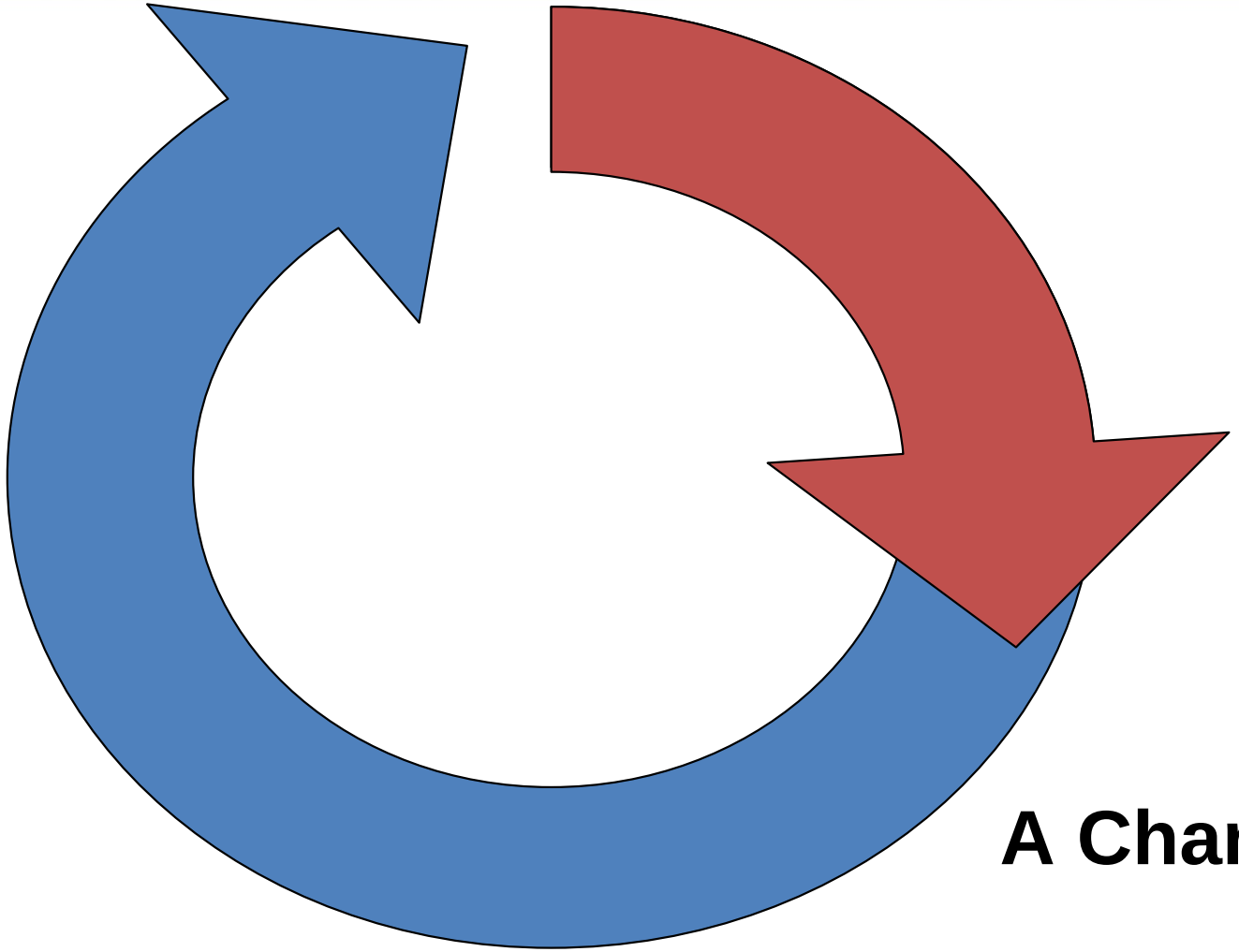
A Spin Around the Wheel



A Full Spin



Maybe a Little More?

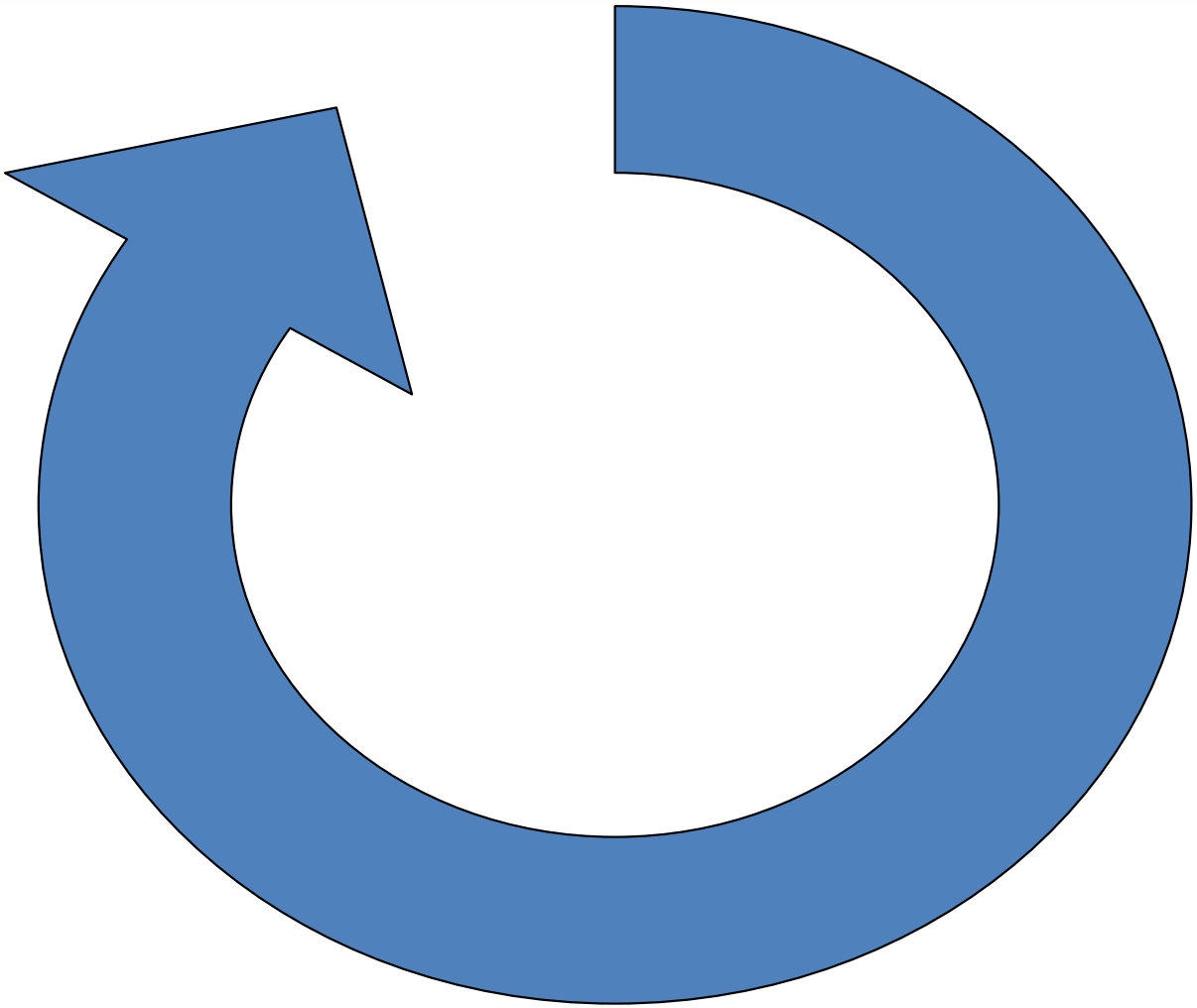


A Channel?

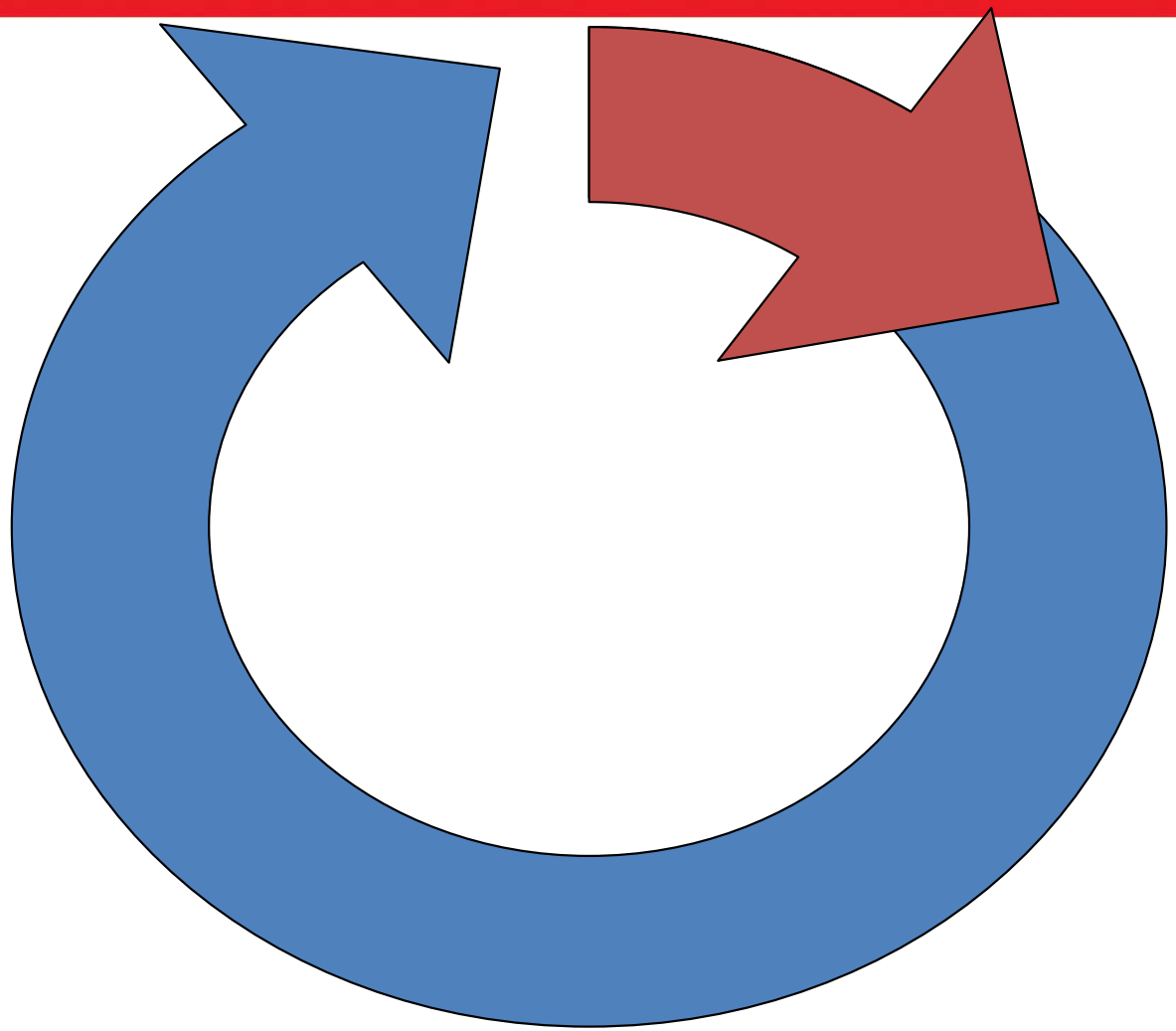
State of the Art Today

- Commercial, high-end GPUs (such as from NVIDIA and ATI) have become very general
  - Unified shader
  - Some special purpose circuits: texturing, rasterization, blending, etc.
- Forthcoming Larrabee from Intel even more general purpose
  - Only special purpose circuit is for texturing
- Where are we on the wheel?

Existing Chips



Near Future (Larrabee paper, S08)?



Enablers for Throughput Computing

- ✓ High level language programming
- ✓ Excellent cost performance
- ✓ Rapid increases in performance over time
- ✓ High memory bandwidth
- ✓ General purpose

Ready for use!

Schedule of Presentations

8:45 Throughput Computing: Hardware Basics

Justin Hensley

9:30 Introduction to Parallel Programming Models

Tim Foley

10:15 Break

Schedule of Presentations

10:30 Introduction to CUDA

Mark Harris

11:00 BSGP: Bulk-Synchronous GPU Programming

Kun Zhou

11:30 OpenCL

Jason Yang

12:15 Lunch

Schedule of Presentations

13:45 Real-Time Reyes: Programmable Pipelines and Research Challenges

Anjul Patney

14:15 Parallel Programming on Larrabee

Tim Foley

14:50 Stream Computing for Graphics

Jeremy Shopf

15:30 Break

Schedule of Presentations

15:45 Parallel Geometry Processing on Graphics Hardware

Pedro V. Sander

16:10 Computational Graphics and Physics Simulation with CUDA

Mark Harris

16:50 Next-Generation Graphics on Larrabee

Tim Foley

17:25 Conclusion and Final Questions

Thank You!

SIGGRAPHASIA2008
NEW HORIZONS