EN 600.320/420 Parallel Programming

General Purpose Computing on the GPU

Slides Originally Prepared By

- Matthew Bolitho

  Then: PhD student, Computer Graphics Lab, Johns Hopkins University

  Later: Manager of Scientific Computing, nVidia

  Later: Directory of Architecture, nVidia

  Now heavily modified as technology evolves

Graphics Processors

- What is a GPU?

Graphics Processing Unit

- Specialized hardware for rendering 3D graphics

NVIDIA GTX 1080

Graphics Processors

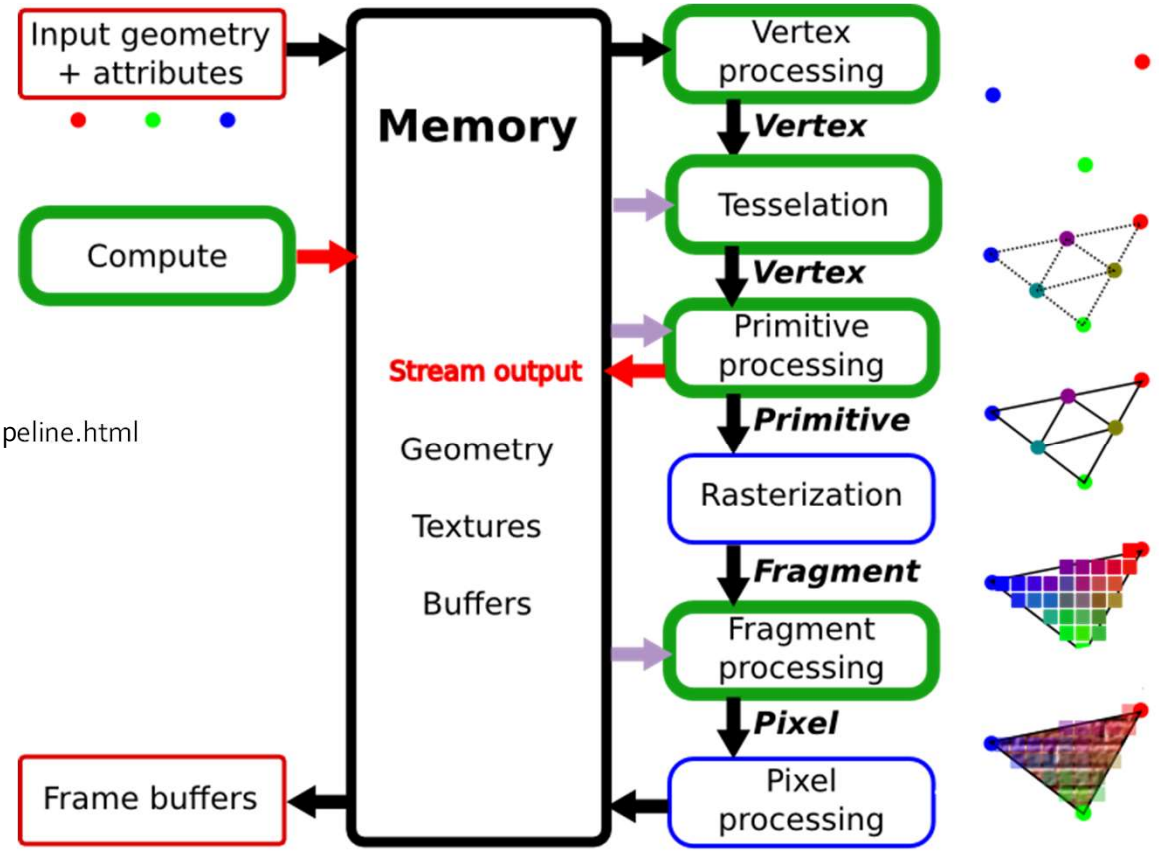
- GPU's are highly parallel processors
  - In rendering, vertices and pixels can be processed in parallel
- GPU's are programmable

Graphics Pipeline

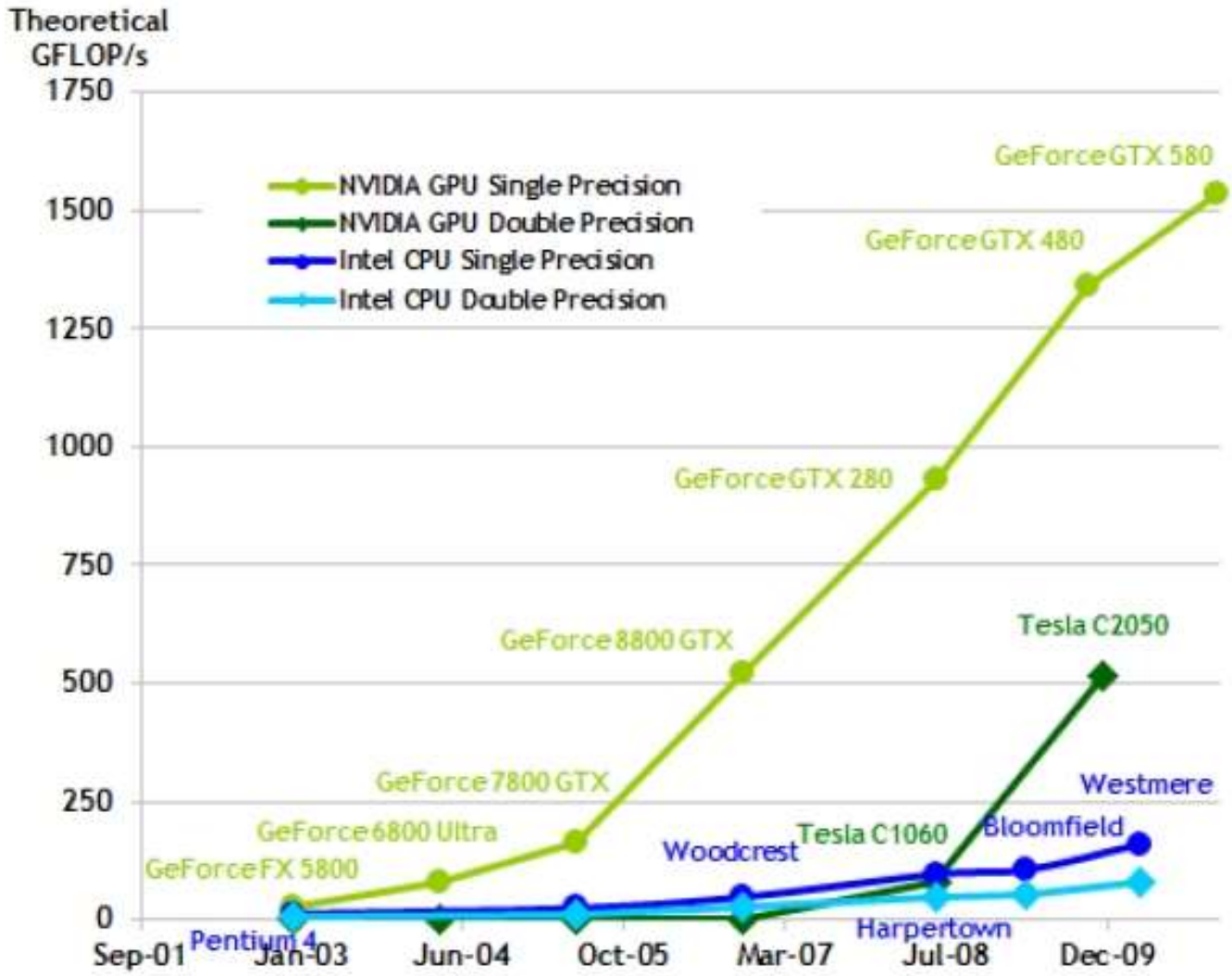
Evolution from fixed function/custom hardware to prog rammable function that reuses multicore arrays

OpenGL pipeline

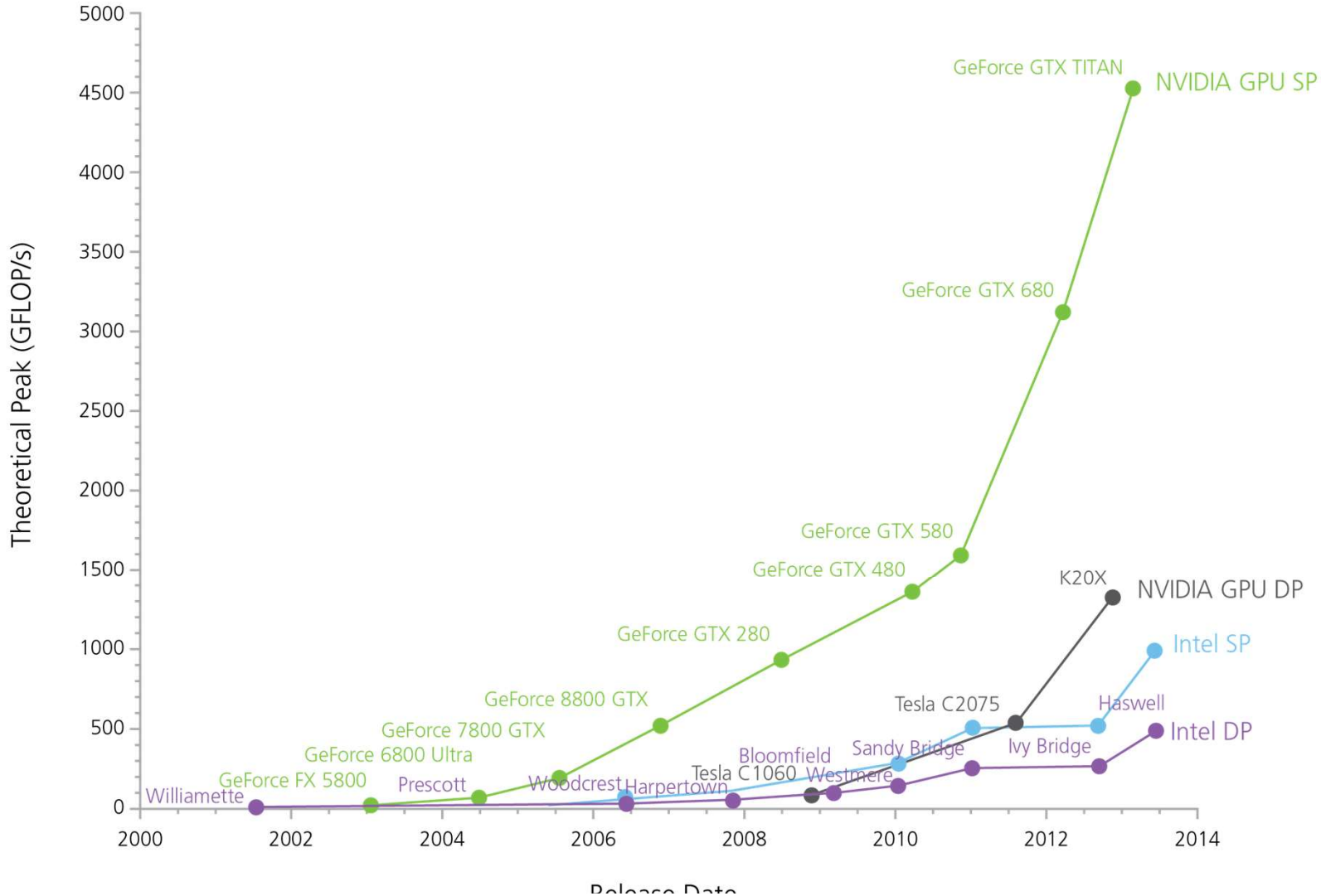
http://romain.vergne.free.fr/teaching/IS/SI03-pipeline.html



Why use GPU's for Compute?

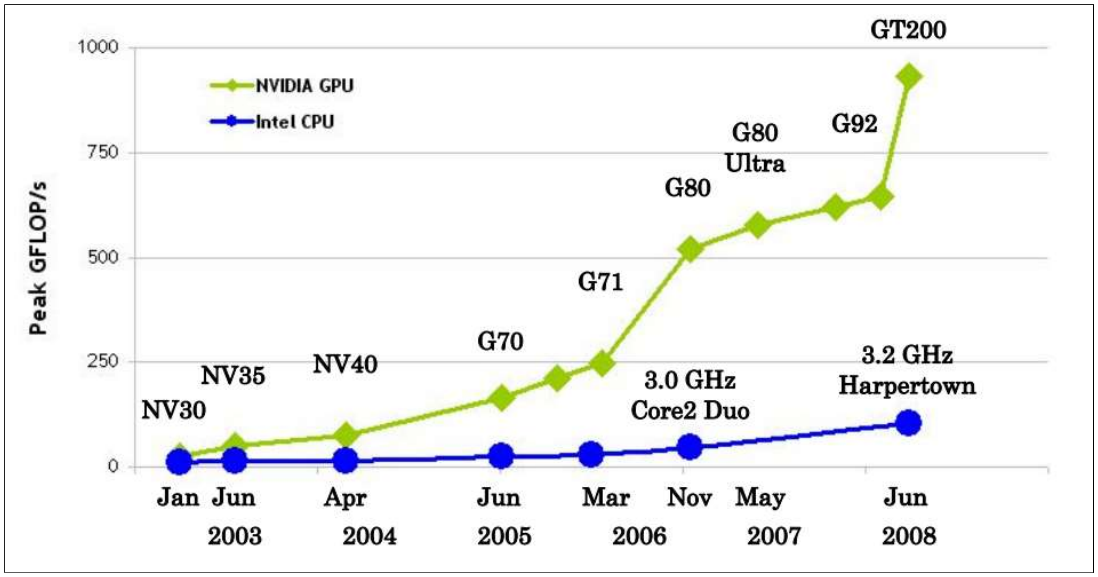


Why use GPU's for Compute?



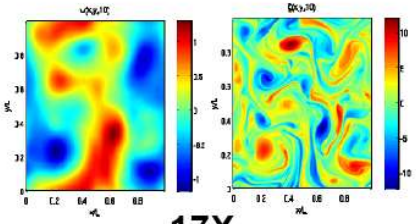
Why use GPU's for Compute?

Product	Memory Bandwidth	Peak Flops
Intel Quad Core CPU	6.4GB/s	63 GFLOPS
NVIDIA GeForce G80	103GB/s	350 GFLOPS
NVIDIA GeForce GT200	180GB/s	1.2 TFLOPS

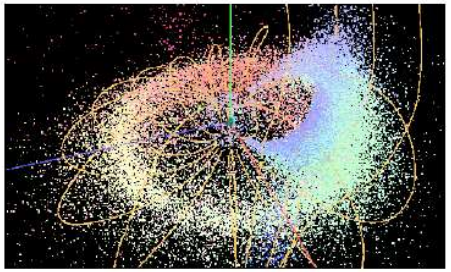


Why use GPU's for Compute?

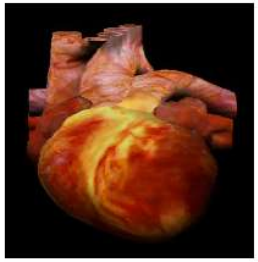
45X



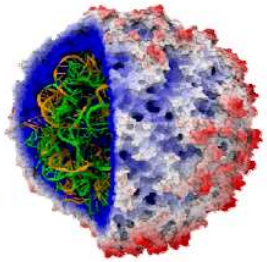
17X



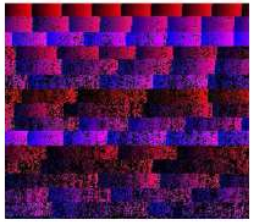
100X



40-170X



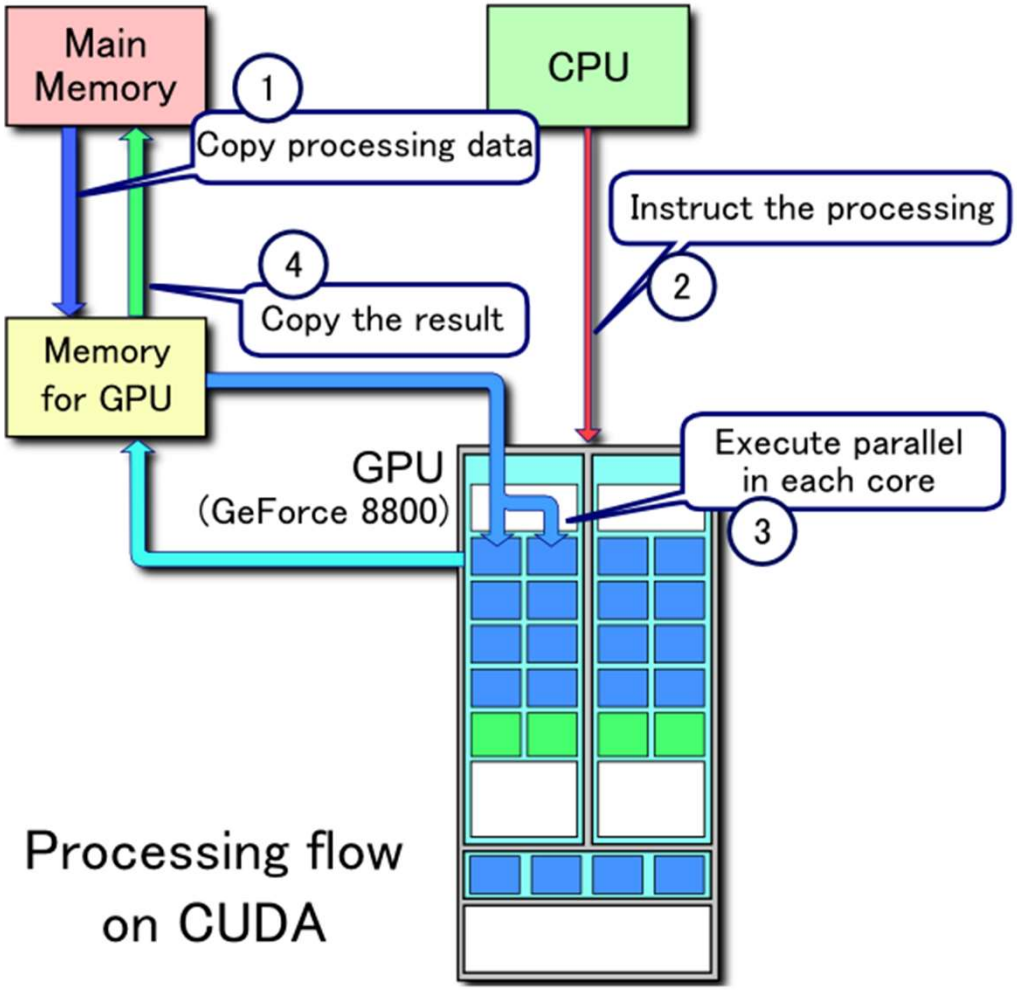
110X



35X

Figure courtesy of Dave Luebke, SC07 CUDA Course

Computing with GPU 101



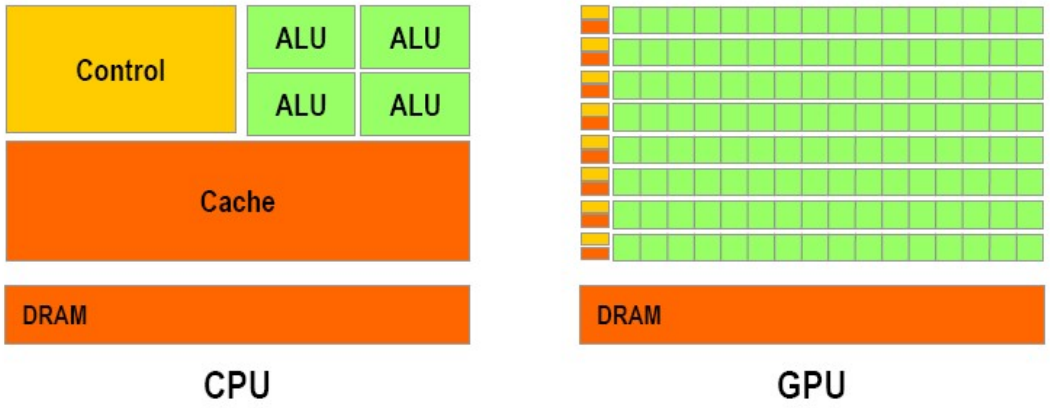
Why so fast?

- Designed for math-intensive, parallel problems:

- More transistors dedicated to ALU than flow control and data cache

Why so fast?

- What are the consequences?



Is it free?

- What are the consequences?
- Program must be more predictable:
  - Data access coherency
  - Program flow