Introduction to GPUs for HPC

High Performance Computing

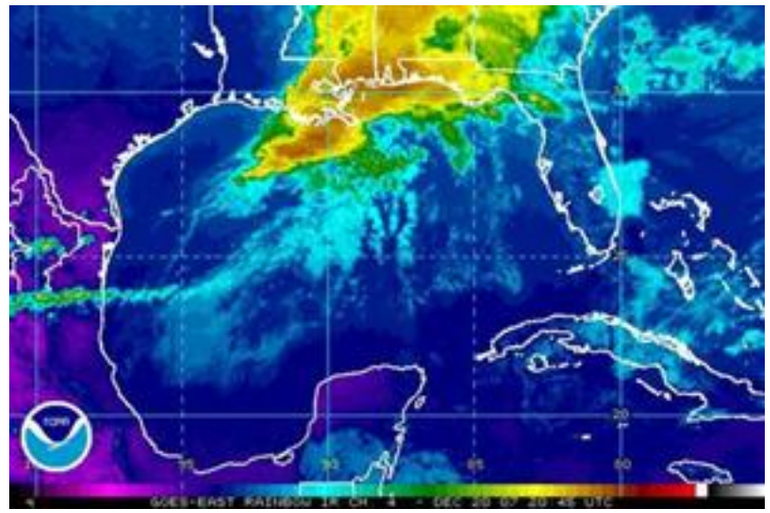
- Speed: Many problems that are interesting to scientists and engineers would take a long time to execute on a PC or laptop: months, years, "never".

- Size: Many problems that are interesting to scientists and engineers can't fit on a PC or laptop with a few GB of RAM or a few 100s of GB of disk space.

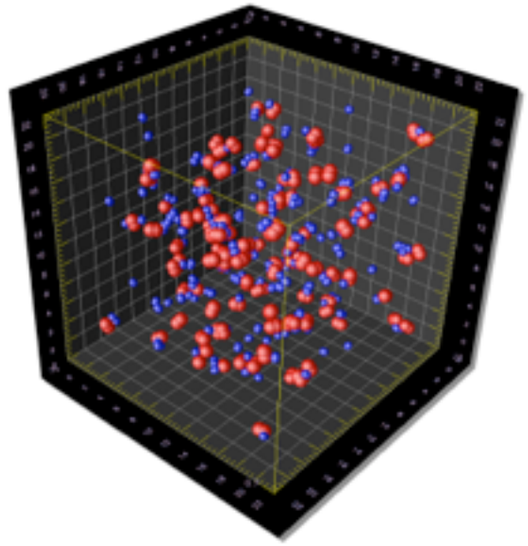
- Supercomputers or clusters of computers can make these problems practically numerically solvable.

Scientific and Engineering Problems

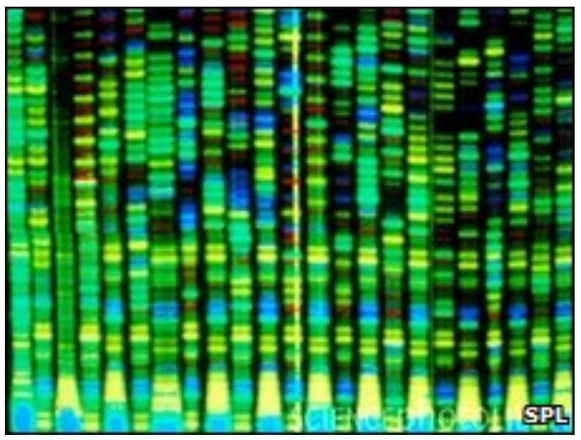
- Simulations of physical phenomena such as:
  - Weather forecasting
  - Earthquake forecasting
  - Galaxy formation
  - Oil reservoir management
  - Molecular dynamics



- Data Mining: Finding needles of critical information in a haystack of data such as:
  - Bioinformatics
  - Signal processing
  - Detecting storms that might turn into hurricanes



- Visualization: turning a vast sea of data into pictures that scientists can understand.



- At its most basic level, all of these problems involve many, many **floating point operations**.

Hardware Accelerators

- In HPC, an accelerator is hardware component whose role is to speed up some aspect of the computing workload.
- In the olden days (1980s), supercomputers sometimes had ***array processors***, which did vector operations on arrays
- PCs sometimes had ***floating point accelerators***: little chips that did the floating point calculations in hardware rather than software.

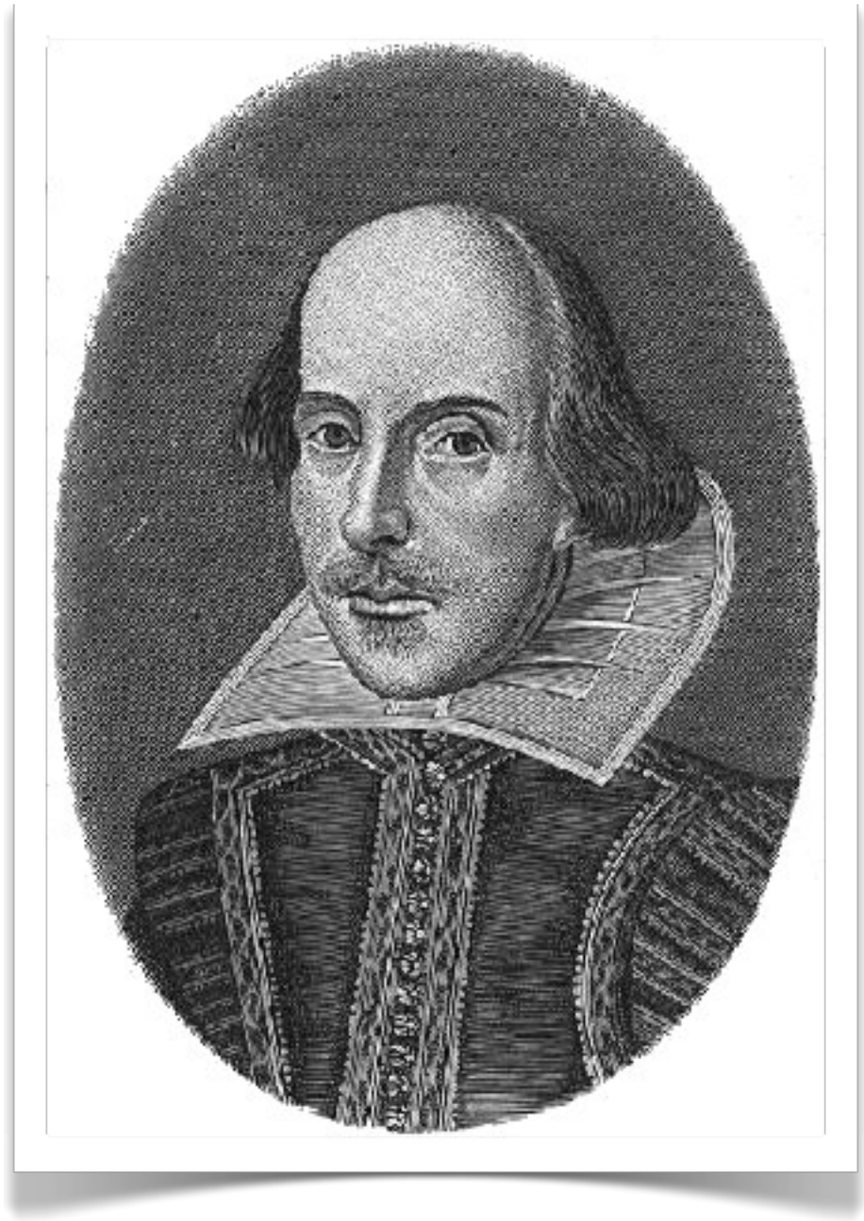
Not an accelerator*


Not an accelerator*

*Okay, I lied.

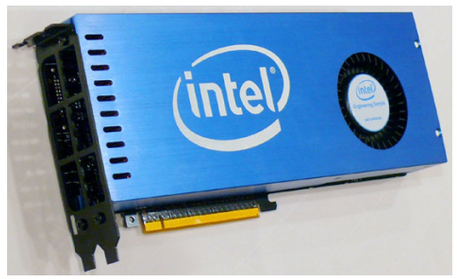
To Accelerate Or Not To Accelerate

- Pro:
  - They make your code run faster.
- Cons:
  - They're expensive.
  - They're hard to program.
  - Your code may not be cross-platform.



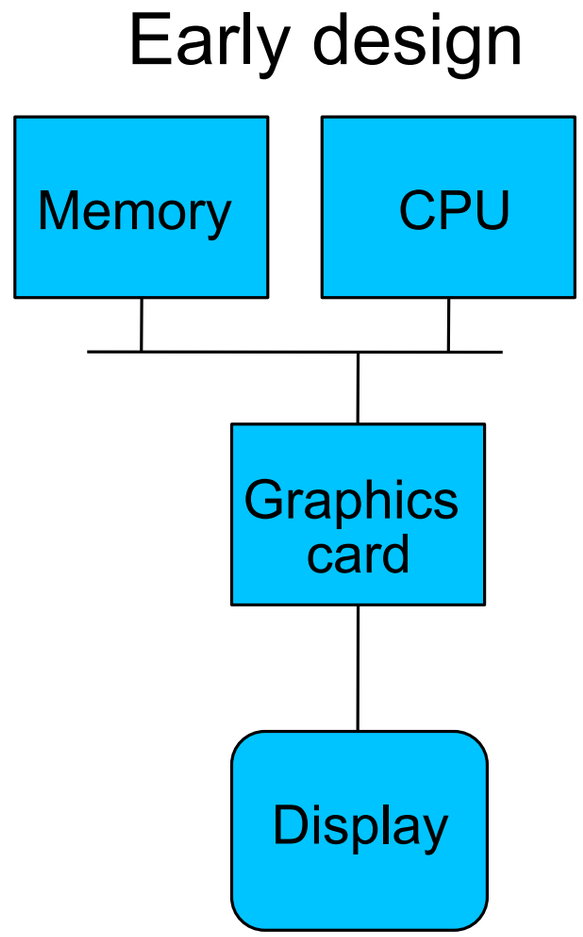
Why GPU for HPC?

- ***Graphics Processing Units*** (GPUs) were originally designed to accelerate graphics tasks like image rendering.
- They became very very popular with video gamers, because they've produced better and better images, and lightning fast.
- And, prices have been extremely good, ranging from three figures at the low end to four figures at the high end.
- Chips are expensive to design (hundreds of millions of $$$), expensive to build the factory for (billions of $$$), but cheap to produce.
- For example, in 2006 – 2007, GPUs sold at a rate of about 80 million cards per year, generating about $20 billion per year in revenue.
- This means that the GPU companies have been able to recoup the huge fixed costs.
- Remember: GPUs mostly do stuff like rendering images. This is done through mostly floating point arithmetic – the same stuff people use supercomputing for!



What are GPUs?

- GPUs have developed from graphics cards into a platform for high performance computing (HPC) -- perhaps the most important development in HPC for many years.
- Co-processors -- very old idea that appeared in 1970s and 1980s with floating point co-processors attached to microprocessors that did not then have floating point capability.
- These coprocessors simply executed floating point instructions that were fetched from memory.
- Around same time, interest to provide hardware support for displays, especially with increasing use of graphics and PC games.
- Led to graphics processing units (GPUs) attached to CPU to create video display.

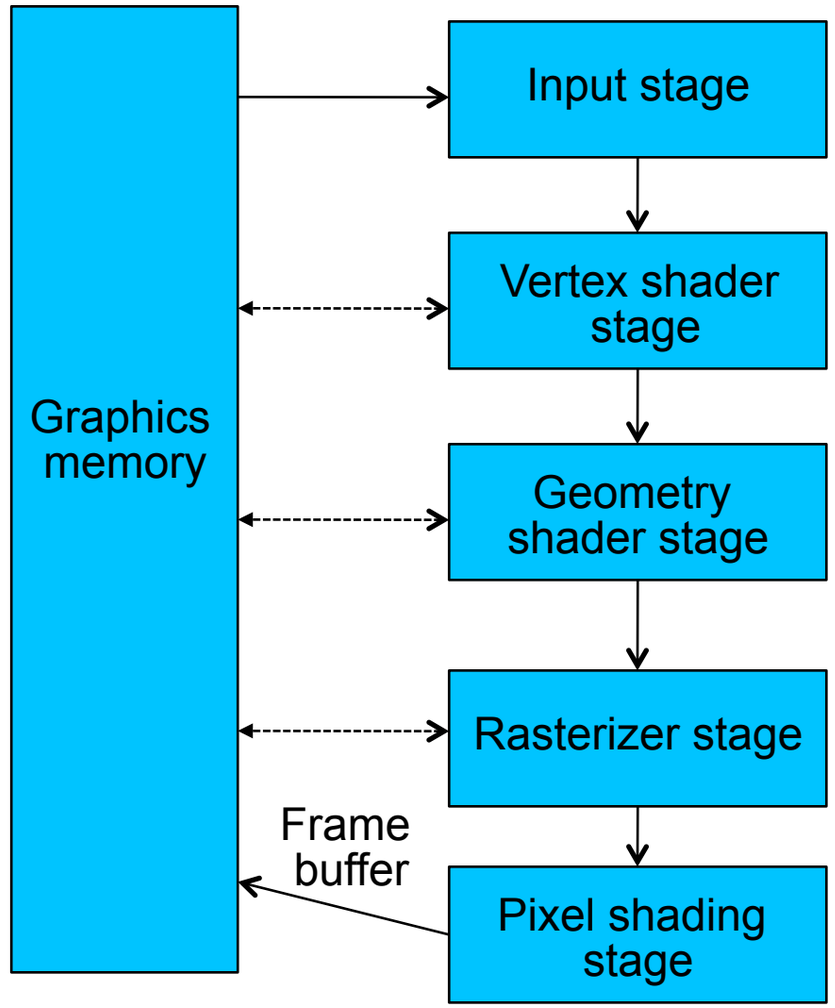


Modern GPU Design

- By late 1990's, graphics chips needed to support 3-D graphics, especially for games and graphics APIs such as DirectX and OpenGL.

- Graphics chips generally had a pipeline structure with individual stages performing specialized operations, finally leading to loading frame buffer for display.

- Individual stages may have access to graphics memory for storing intermediate computed data.

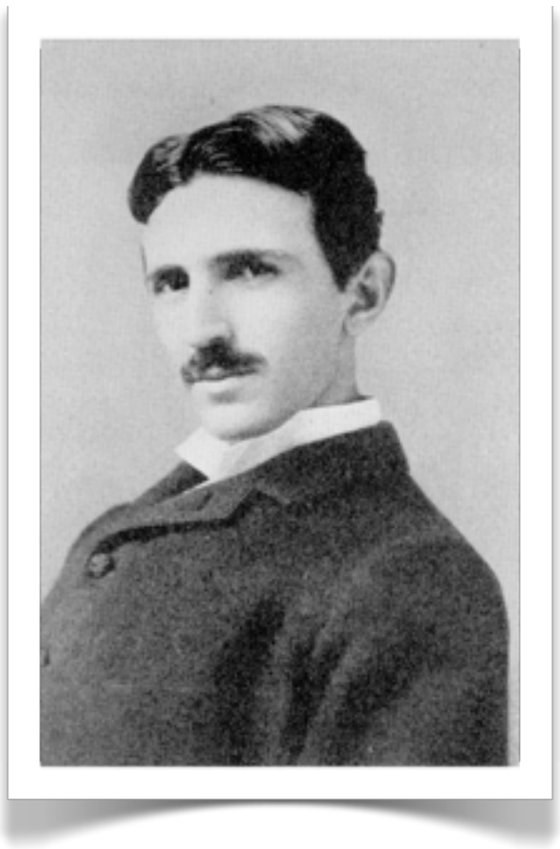


General Purpose GPU (GPGPU) Designs

- High performance pipelines call for high-speed (IEEE) floating point operations.
- Known as GPGPU (General-purpose computing on graphics processing units) -- Difficult to do with specialized graphics pipelines, but possible.)
- By mid 2000's, recognized that individual stages of graphics pipeline could be implemented by a more general purpose processor core (although with a data-parallel paradigm)
- 2006 -- First GPU for general high performance computing as well as graphics processing, NVIDIA GT 80 chip/GeForce 8800 card.
- Unified processors that could perform vertex, geometry, pixel, and **general computing operations**
- Could now write programs in C rather than graphics APIs.
- Single-instruction multiple thread (SIMT) programming model

NVIDIA Tesla Platform

- NVIDIA Tesla series was their first platform for the high performance computing market.
- Named for Nikola Tesla, a pioneering mechanical and electrical engineer and inventor.



GTX 480

C2070

NVIDIA GTX 480 Specs

- 3 billion transistors
- 480 compute cores
- 1.401 GHz
- Single precision floating point performance: 1.35 TFLOPs (2 single precision flops per clock per core)
- Double precision floating point performance: 168 GFLOPs (1 double precision flop per clock per core)
- Internal RAM: 1.5 GB DDR5 VRAM
- Internal RAM speed: 177.4 GB/sec (compared 21-25 GB/sec for regular RAM)
- PCIe slot (at most 8 GB/sec per GPU card)
- 250 W thermal power

Coming: Kepler and Maxwell

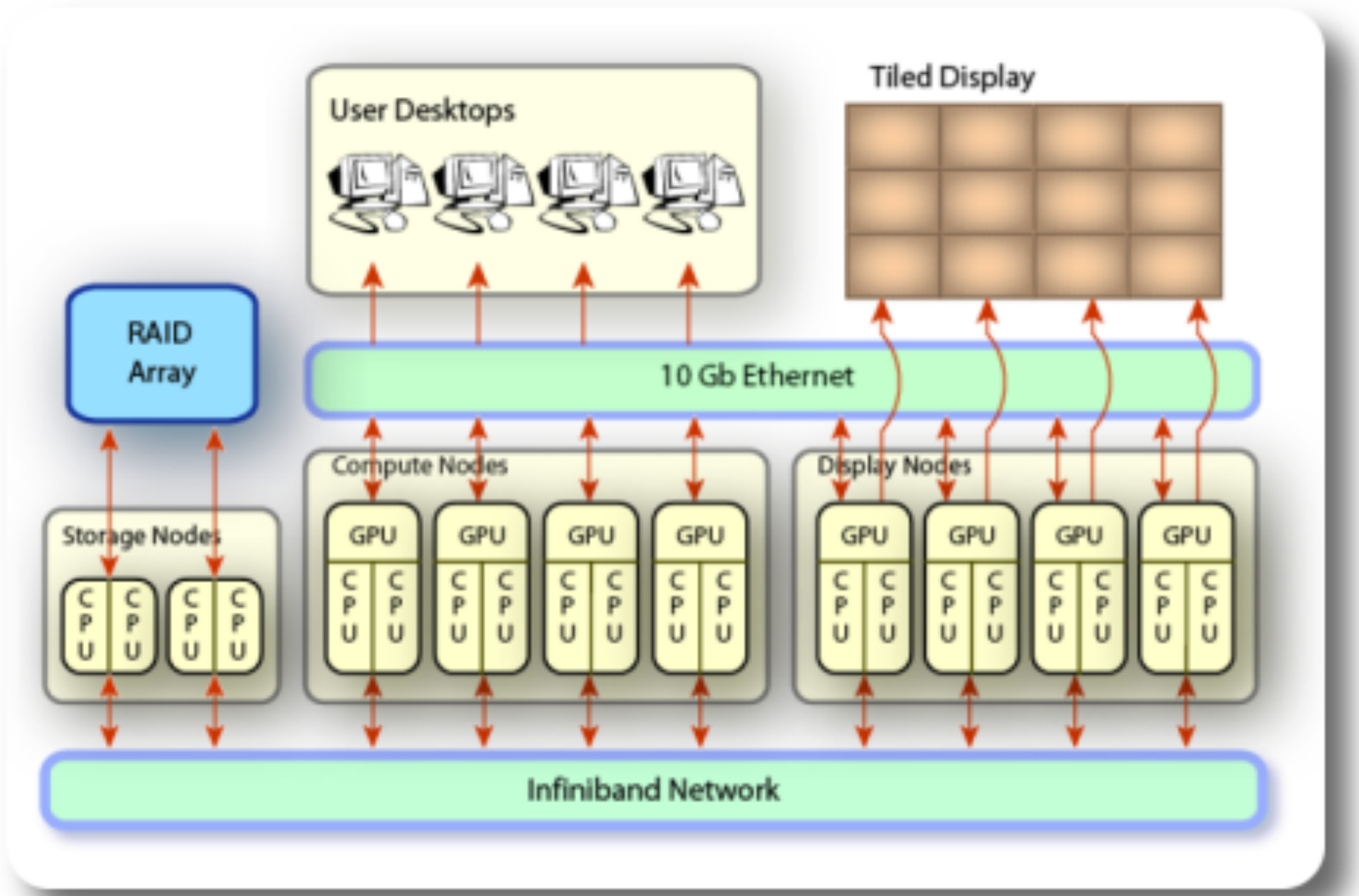
- NVIDIA's 20-series is also known by the codename "Fermi." It runs at about 0.5 TFLOPs per GPU card (peak).

- The next generation, to be released in 2011, is codenamed "Kepler" and will be capable of something like 1.4 TFLOPs double precision per GPU card.

- After "Kepler" will come "Maxwell" in 2013, capable of something like 4 TFLOPs double precision per GPU card.

- So, the increase in performance is likely to be roughly 2.5x – 3x per generation, roughly every two years.

Maryland CPU/GPU Cluster Infrastructure



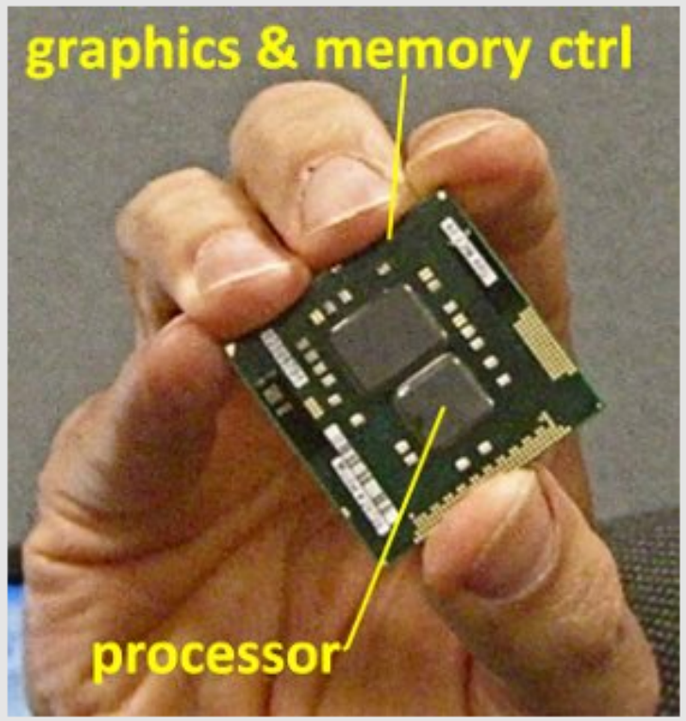
http://www.umiacs.umd.edu/
research/GPU/facilities.html

Intel's Response to NVIDIA GPUs

iSGTW INTERNATIONAL SCIENCE GRID THIS WEEK

Feature - GPU-based cheap supercomputing coming to an end

Nvidia's CUDA has been hailed as "Supercomputing for the Masses," and with good reason – amazing speedups ranging from 10x through hundreds have been reported on scientific / technical code. CUDA has become a darling of academic computing and a major player in DARPA's Exascale program, but performance alone does not account for that popularity: price clinches the deal. For all that computing power, they're incredibly cheap. As Sharon Glotzer of UMich noted, "Today you can get two gigaflops for $500. That is ridiculous." It is indeed. And it's only possible because CUDA is subsidized by sinking the fixed costs of its development into the high volumes of Nvidia's mass market low-end GPUs.



Intel's Sandy Bridge architecture places the processor and GPU on the same chip.

Image courtesy Greg Pfister.

Unfortunately, that subsidy won't last forever; its end is now visible. Intel has now started pounding the marketing drums on something long predicted: integration of Intel's graphics onto the same die as its next generation "Sandy Bridge" processor chip, due out in mid-2011.

Probably not coincidentally, mid-2011 is when AMD's Llano processor will see daylight. It incorporates enough graphics-related processing to be an apparently decent DX11 GPU, although to my knowledge the architecture hasn't been disclosed in detail.

Just prior to this Fall's IDF (Intel Developer Forum), Anandtech received an early demo part of Sandy Bridge and checked out the graphics, among other things. Their net is that

Does it work?

Example Applications	URL	Speedup
Seismic Database	http://www.headwave.com	66x – 100x
Mobile Phone Antenna Simulation	http://www.accelware.com	45x
Molecular Dynamics	http://www.ks.uiuc.edu/Research/vmd	21x – 100x
Neuron Simulation	http://www.evolvedmachines.com	100x
MRI Processing	http://bic-test.beckman.uiuc.edu	245x – 415x
Atmospheric Cloud Simulation	http://www.cs.clemson.edu/~jesteel/clouds.html	50x

- Looks like remarkable speedup compared to traditional CPU-based HPC approaches