



Information Coding / Computer Graphics, ISY, LiTH

**And this brought us to:
GPGPU/GPU Computing**

General Purpose computation on Graphics Processing Units

Mark Harris, 2002

Perform demanding calculations on the GPU instead of the CPU!

At first, appeared to be a wild idea, but is now a very serious technology! Results were highly varied in the early years, but the GPU advantage has grown bigger and bigger.



Key components of the GPGPU trend

High processing power in parallel

Programmability: Introduction of shader programs, much more flexible, programmable for any problem.

Floating-point buffers: Vital! Initially with poor precision. 32-bit floating-point decent... but not really impressive.



GPGPU approaches

- **Using fixed pipeline graphics**
 - **Shader programs**
 - **CUDA**
 - **OpenCL**
 - **Compute shaders**



Fixed pipeline GPGPU

Reformulate a problem to something that can be done by standard graphics operations.

Limited success 1999/2000. Not of any practical interest!

Example: Jörgen Ahlberg, face tracking



Shader-based GPGPU

Portable! All GPUs can use shaders, no need for extra software, run using standard software/drivers.

All modern shader languages (GLSL, Cg, HLSL) are similar and easy to program in.

Requires a re-mapping of data to textures.

Very good results already in 2005: 8x speedups overall reported!



CUDA-based GPGPU

Only works on NVidia hardware.

Requires extra software - which isn't very elegant.

Nice integration of CPU and GPU code in the same program.

Excellent results! 100x speedups are common - before optimizing! Even low-end GPUs give significant boosts.



OpenCL-based GPGPU

Works on various hardware - not only GPUs.

Developed by Khronos Group, pushed by Apple.

Harder to get started, software looks pretty much like programming shaders.



Compute shaders

Built into OpenGL

Similar to OpenCL

Good portability



Vulkan

The "new OpenGL", arrived 2016.

"Bleeding edge".

**Future main generic GPU platform for
both graphics and computing?**

Same compute shaders as OpenGL.



Use the source, Luke!

Four trivial examples:

Hello World! for CUDA

Hello World! for OpenCL

Hello World for GLSL

Hello World for Compute Shaders



Information Coding / Computer Graphics, ISY, LiTH

In the Signal&Bild lab

GTX1080

Pascal GPUs, brand new!

2560 cores!

**The new Turing is to of the line, but 1080 is still close to the top
and perfect for lab use at this time.**



Information Coding / Computer Graphics, ISY, LiTH

In the Multicore lab

GTX560Ti

Fermi! Old, but good computing performance!

Being phased out.



Information Coding / Computer Graphics, ISY, LiTH

That's all, folks!

Next time: Introduction to CUDA