GPU: The Ultimate Commodity Supercomputer

Jiří Matela

&

Martin Jirman

COMPRIMATO

jpeg2000@GPU

@comprimato

The Evolution of Computing

Intel ASCI Red

1 TFLOPS

7904 CPUs

850 KW

150 m2

The Evolution of Computing

NVIDIA GeForce

5.1 TFLOPS

250W

296 cm2

The Evolution of Computing

Mobile GPU

0.36 TFLOPS

5W

0.14 cm2

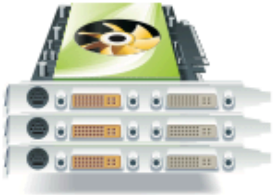
Same performance, different costs

Google brain

1,000 CPU Servers
2,000 CPUs - 16,000 cores

600 kWatts
$ 5,000,000

Standford AI Lab



3 GPU Accelerated Servers
12 GPUs - 18,432 cores

4 kWatts
$ 33,000

Artifical Brain – Neural Network – Deep learning

* Source NVIDIA

Where GPUs Shine

Neural networks (Netflix)

GPU accelerated database query (PgOpenCL)

Physics (Games)

Ray tracing (FurryBall, NVIDIA OptiX)

Linear Algebra (CUBLAS)

Video Compression

General-purpose computing on graphics processing units (GPU)

- Video Controller -> GPU (nvidia) -> GPGPU
- Shaders Languages -> CUDA / OpenCL
- CUDA
  - Computing architecture
  - Programming language

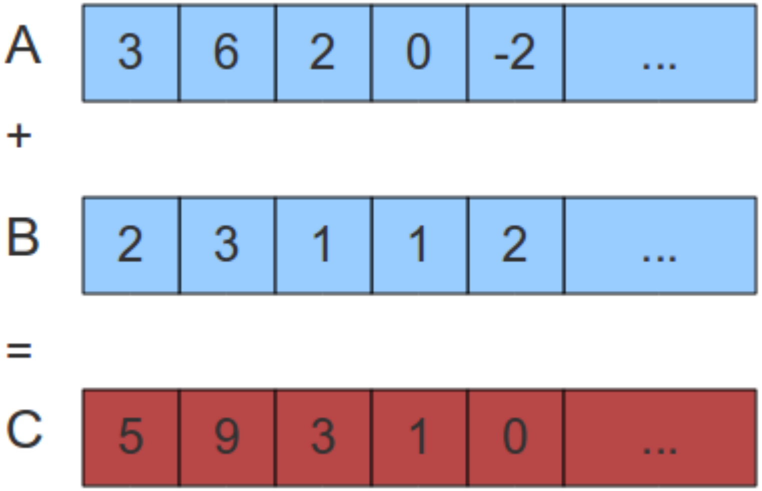
CUDA Quick Start SLIDE

- NVIDIA GPU – GeForce (mobile), Tesla, Quadro
- Win / Lin / Mac
- NVIDIA Driver
- NVIDIA Installer*
  - Toolkit
  - Samples
  - Tools

* developer.nvidia.com/cuda-downloads

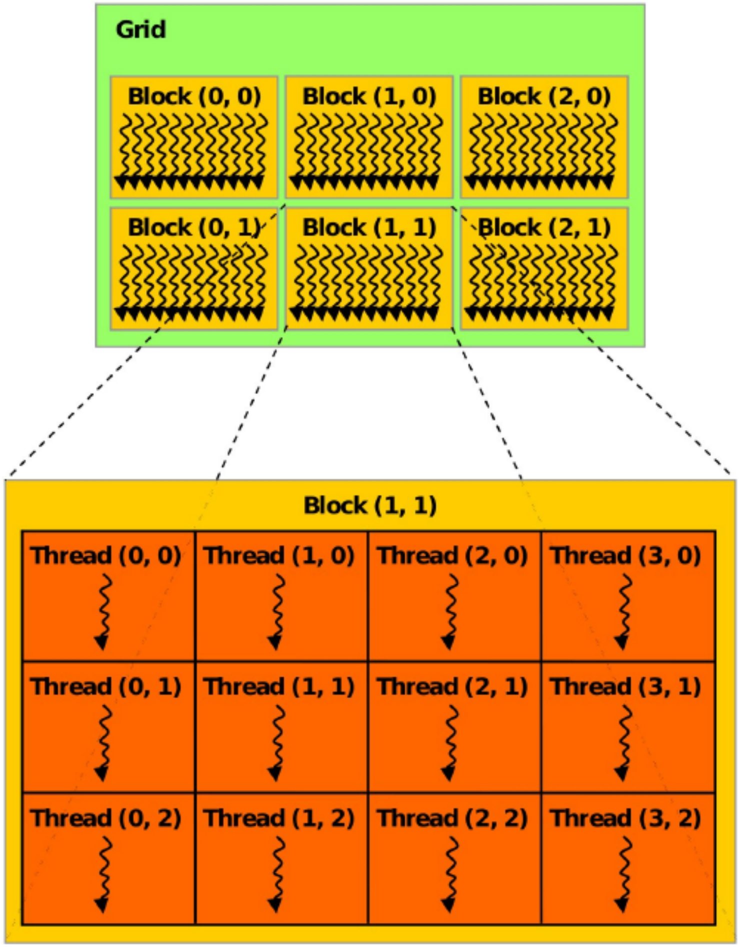
CUDA Architecture and Programming model

1st example – Vector Addition



A	3	6	2	0	-2	...
+						
B	2	3	1	1	2	...
=						
C	5	9	3	1	0	...

Thread Hierarchy



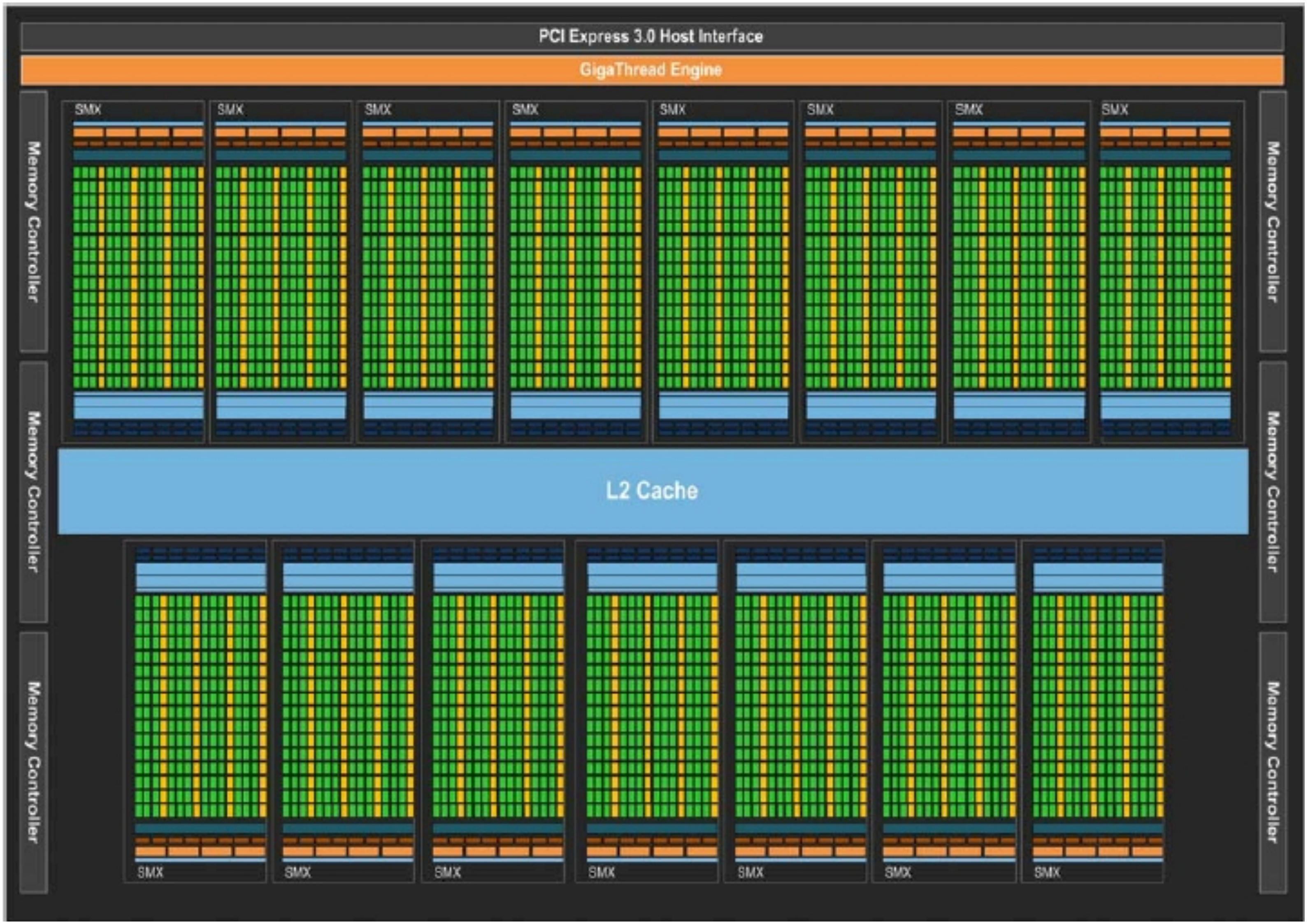
Grid

Block (0, 0)	Block (1, 0)	Block (2, 0)
Block (0, 1)	Block (1, 1)	Block (2, 1)

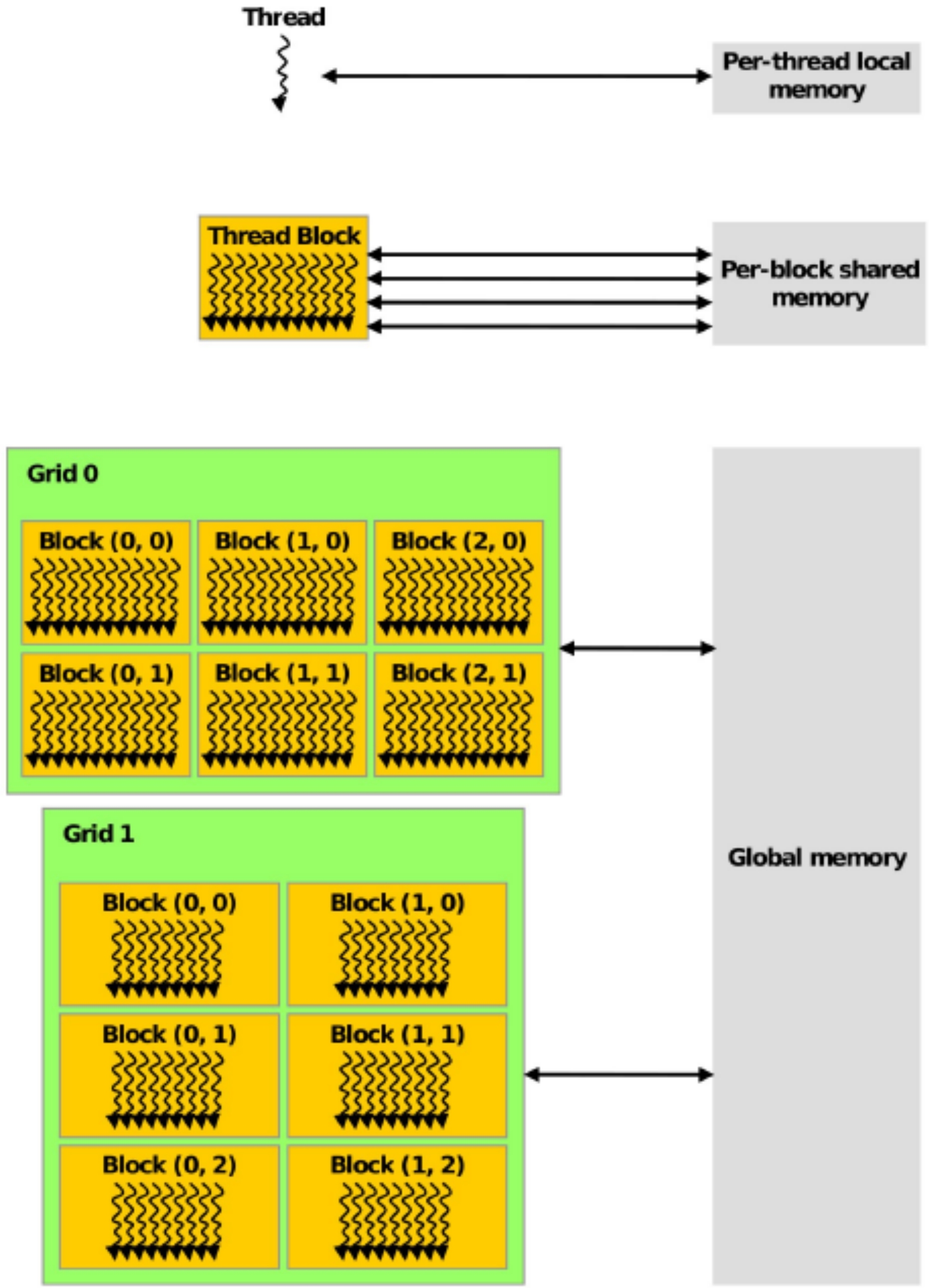
Block (1, 1)

Thread (0, 0)	Thread (1, 0)	Thread (2, 0)	Thread (3, 0)
Thread (0, 1)	Thread (1, 1)	Thread (2, 1)	Thread (3, 1)
Thread (0, 2)	Thread (1, 2)	Thread (2, 2)	Thread (3, 2)

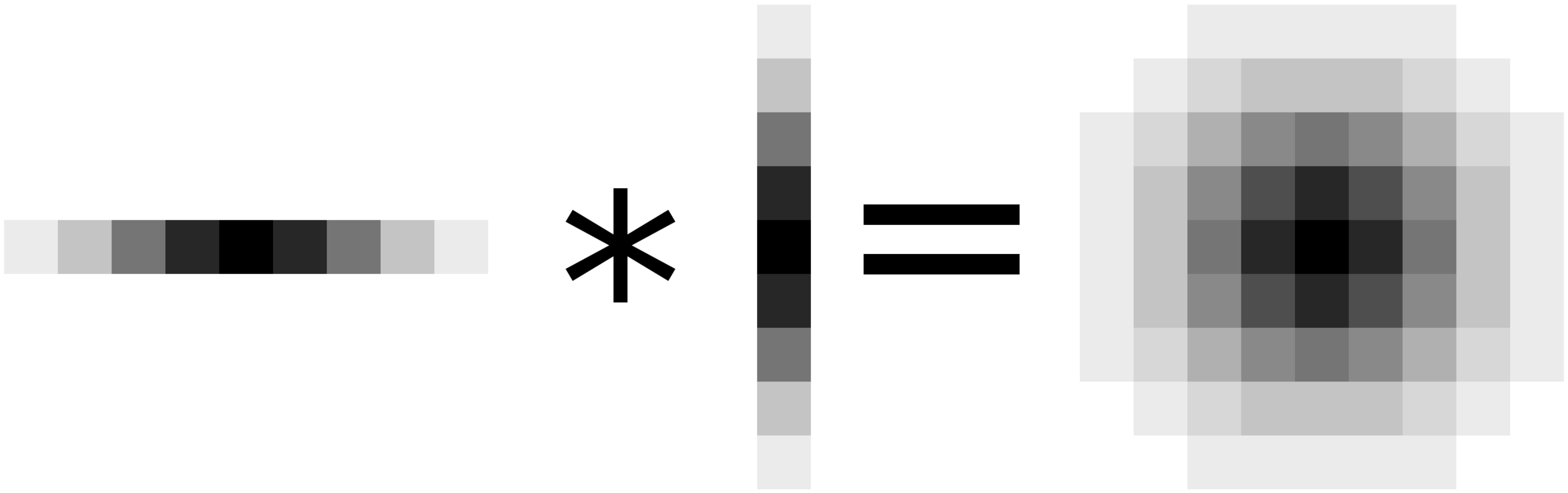
GPU as Multicore SIMD



Memory Hierarchy



Gaussian Image Blur



CPU Basic Single Thread Implementation

GPU Basic Implementation
(1pixel = 1thread)

CPU Parallel Using OpenMP

GPU using Shared Memory

GPU – Overlapping Transfers and Computations

GPU – Final

1 thread = multiple pixels

private array (registers)

#pragma unroll

Conclusion

- Gaussian blur
- CPU 160 -> 100 ms
  - Core i5 – 4 Cores
- GPU 16 -> 3ms
  - GeFroce 740m
  - 2 SM
  - 368 cuda cores

Thank you!

Jiří Matela & Martin Jirman

COMPRIMATO