中国科学院国家天文台
National Astronomical Observatories, CAS

千人计划
RECRUITMENT
PROGRAM OF GLOBAL EXPERTS

UNIVERSITÄT
HEIDELBERG
Zukunft. Seit 1386.

Introduction to GPU Accelerated Computing:
1. History of Computer Architecture
Many-Core, GPU, and other ideas...

University

Rainer Spurzem
Astronomisches Rechen-Inst., ZAH, Univ. of Heidelberg, Germany
National Astronomical Observatories (NAOC), Chinese Academy of Sciences
Kavli Institute for Astronomy and Astrophysics (KIAA), Peking University

spurzem@nao.cas.cn
http://silkroad.bao.ac.cn

the SILK ROAD PROJECT at NAOC/KIAA
丝绸之路 计划

北京大
PEKING UNIVERS

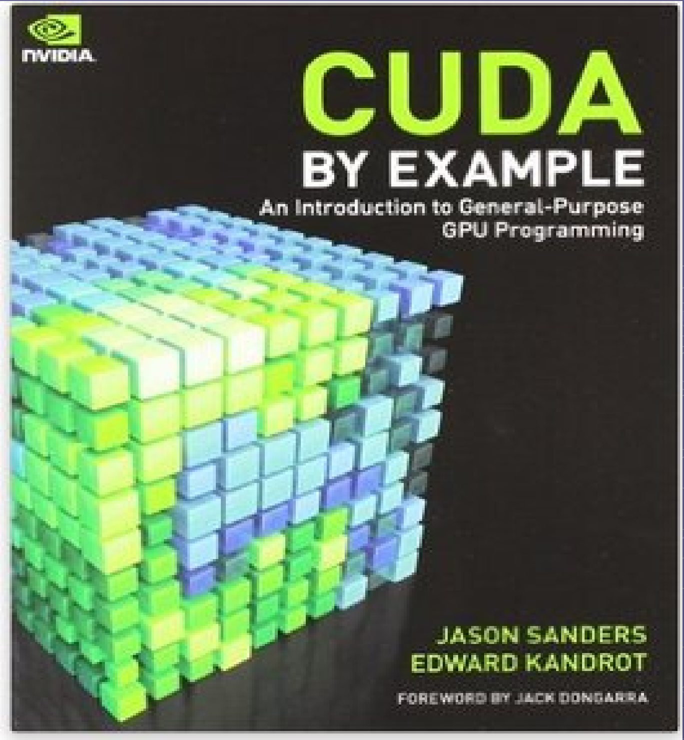
Introduction to GPU Accelerated Computing
March 26 – 29, 2018

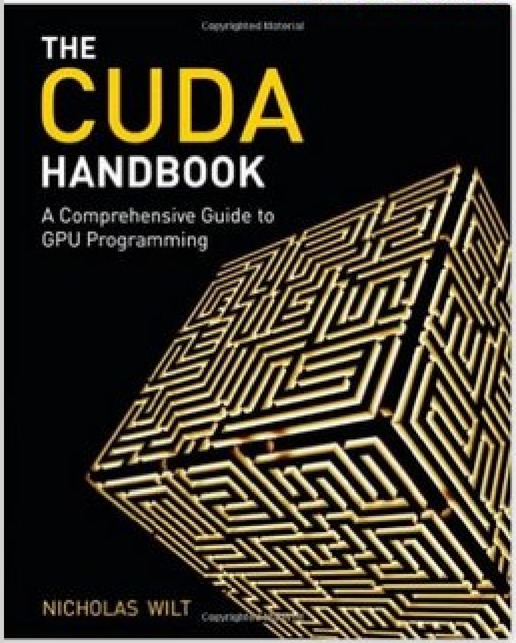
Table of Contents (subject to change):

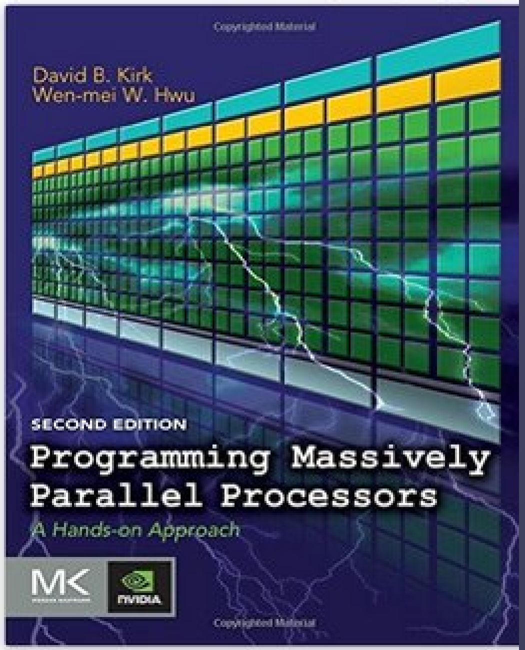
1. Monday morning: General Introduction Computer Architecture, Many-Core, GPU and others…, Access...
2. Monday afternoon: Access to kepler, CUDA Hello, GPU Properties, Simple Add, Vector Add
3. Tuesday morning: More on GPU Software and Hardware
4. Tuesday afternoon: CUDA More Vector Add, Scalar Product, Histograms, Events
5. Wednesday morning: New Features of Kepler Architecture, Astrophysical N-Body Code
6. Wednesday Afternoon: Astrophysical Parallel N-Body Code Using MPI and GPU
7. Thursday Morning: Parallelization and Amdahl's Law
8. Thursday Afternoon: Amdahl's Law and GPU Acceleration
9. Access: Use **ssh-keygen -t rsa** (give passphrase)

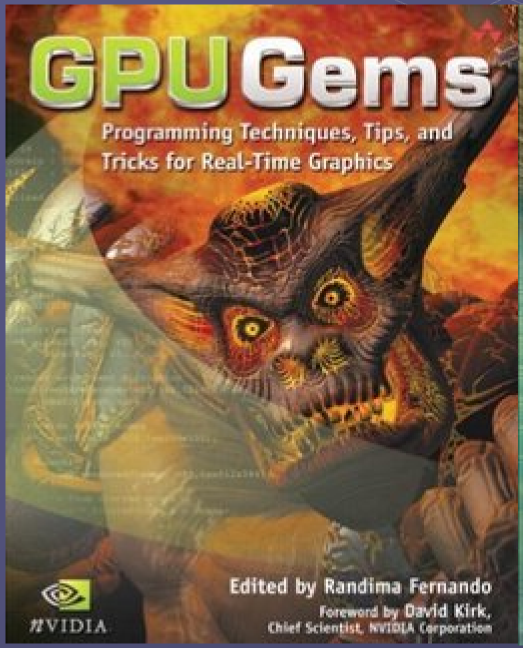
Send **id_rsa.pub** to **spurzem@ari.uni-heidelberg.de**

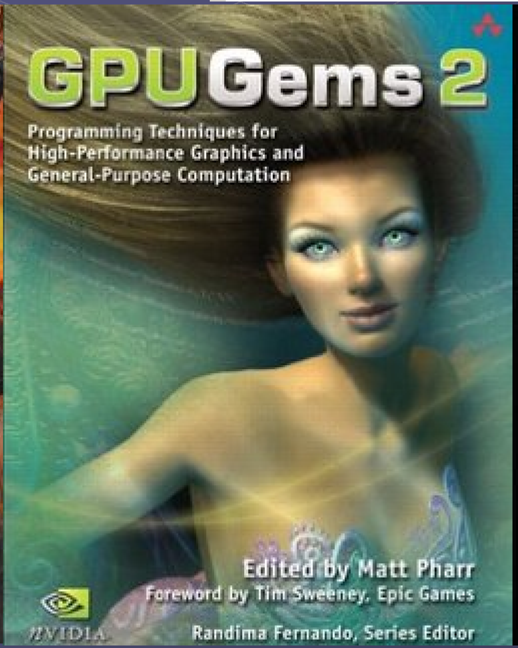
Literature

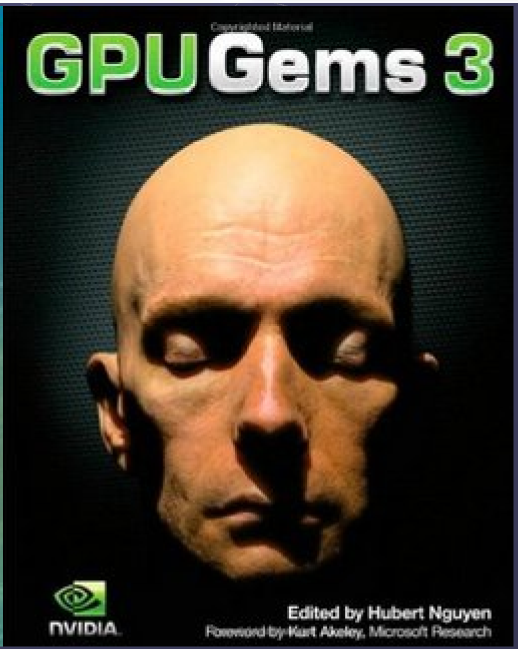

NVIDIA
CUDA
BY EXAMPLE
An Introduction to General-Purpose GPU Programming
JASON SANDERS
EDWARD KANDROT
FOREWORD BY JACK DONGARRA


THE
CUDA
HANDBOOK
A Comprehensive Guide to GPU Programming
NICHOLAS WILT


David B. Kirk
Wen-mei W. Hwu
SECOND EDITION
Programming Massively Parallel Processors
A Hands-on Approach
MK
NVIDIA


GPU Gems
Programming Techniques, Tips, and Tricks for Real-Time Graphics
Edited by Randima Fernando
Foreword by David Kirk, Chief Scientist, NVIDIA Corporation
NVIDIA


GPU Gems 2
Programming Techniques for High-Performance Graphics and General-Purpose Computation
Edited by Matt Pharr
Foreword by Tim Sweeney, Epic Games
Randima Fernando, Series Editor
NVIDIA


GPU Gems 3
NVIDIA
Edited by Hubert Nguyen
Foreword by Kurt Akeley, Microsoft Research

Observations (Experiment)

Theory

Computational Physics

GPU Computing

History

History

Erik Holmberg (1908-2000)

Dissertation Univ. Lund (Schweden) (1937):

``A study of double and multiple galaxies´´

Galaxies often in Groups and Pairs

Irregular Distribution of Satellite Galaxies

(Holmberg-Effect)

Father of numerical astrophysics?

» **...with 200 light bulbs**

History

http://cdsads.u-strasbg.fr/abs/1941ApJ....94..385H

The Astrophysical Journal, Nov. 1941

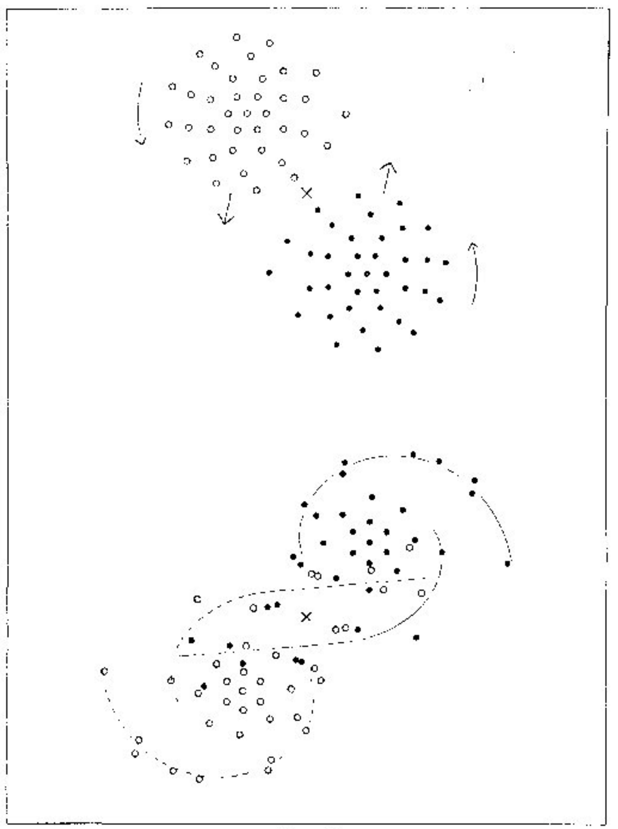


FIG. 4b

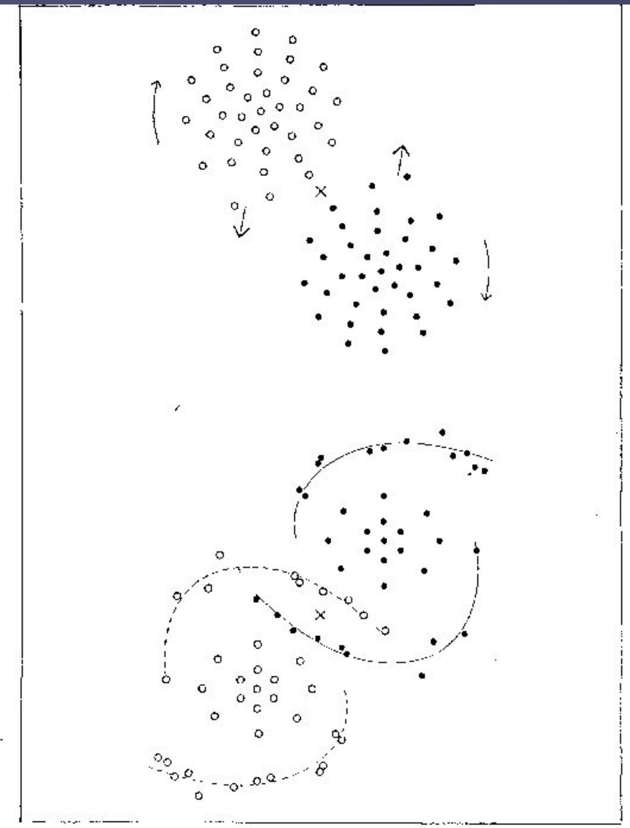


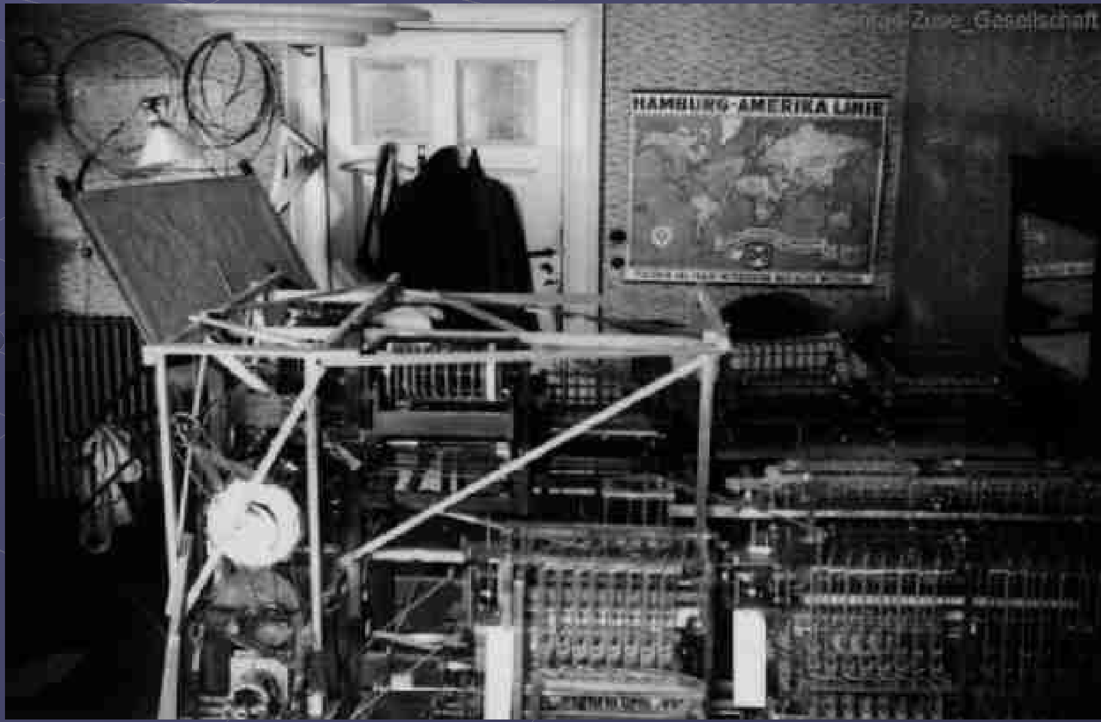
FIG. 4a

HARDWARE

...before von Neumann...

- Konrad Zuse (1910-1995) Berlin

Invented freely programmable Computer



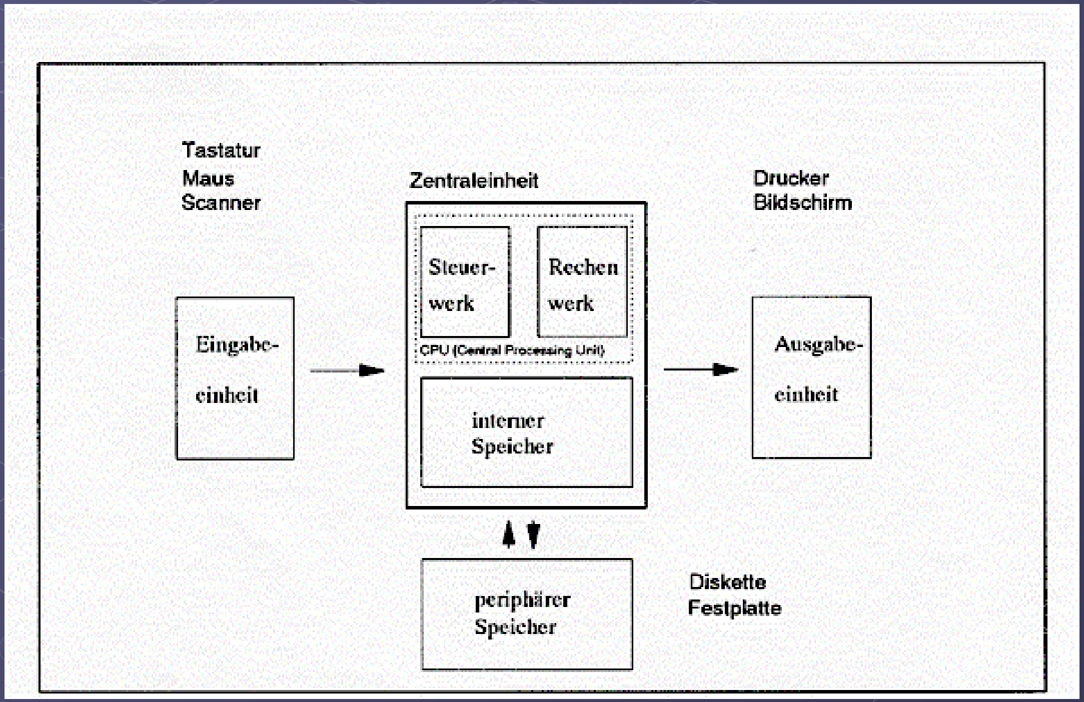
Z1 in parental flat 1936

HARDWARE

- John von Neumann (1903-1957)

Born Budapest, Lecturer Berlin, since 1930 Princeton Univ.

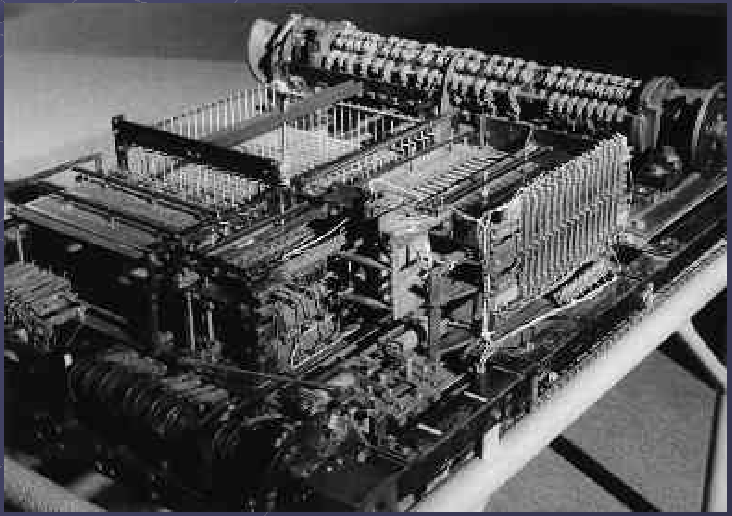
Requirements for the Construction of an electronic computing device(1946)



History

Zuse Z4: 1944 Berlin, 1950 Zürich, 1954 Frankreich

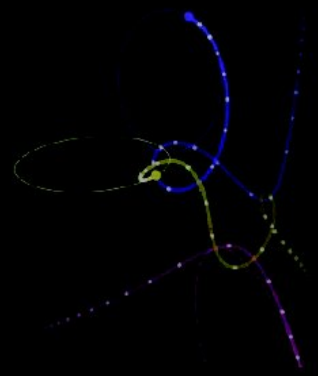
1959 Deutsches Museum München



Computing Speed 0.03 MHz

Memory 256 byte

Astronomisches Rechen-Institut (ARI) at Univ. of Heidelberg, Germany



Siemens 2002 Computer in 1964 At ARI

History

http://cdsads.u-strasbg.fr/abs/1960ZA.....50..184V

Astronomisches Rechen-Institut in Heidelberg
Mitteilungen Serie A Nr. 14

Die numerische Integration des *n*-Körper-Problemes für Sternhaufen I

Von

SEBASTIAN VON HOERNER

Mit 3 Textabbildungen

(Eingegangen am 10. Mai 1960)

Tabelle 5. *Zahl der gegenseitigen Umläufe, Häufigkeit des Auftretens und kleinster gegenseitiger Abstand D_m der engsten Paare.* (Alle engsten Paare mit mehr als zwei vollen Umläufen wurden notiert)

Umläufe	Häufigkeit	D_m
2—3	11	0.0102
3—5	9	0.0177
5—10	5	0.0070
10—20	2	0,0141
20—50	1	0.0007
50—100	1	0.0035
100—200	1	0.0039

Astronomisches Rechen-Institut in Heidelberg
Mitteilungen Serie A Nr. 19

Die numerische Integration des *n*-Körper-Problems für Sternhaufen, II.

Von

SEBASTIAN VON HOERNER

Mit 10 Textabbildungen

(Eingegangen am 19. November 1962)

http://cdsads.u-strasbg.fr/abs/1963ZA.....57...47V

S.v. Hoerner,
Z.f.Astroph. 1960, 63

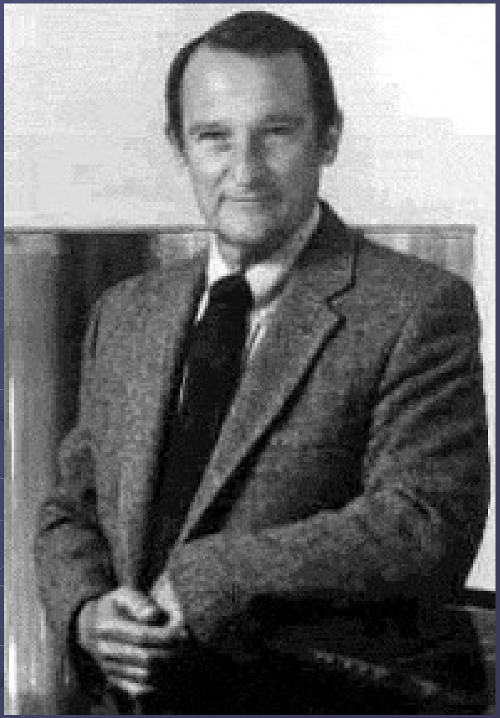
Siemens 2002
N=4,8,12,16 (4 Trx)

N=16,25 (40 Trx)

History

- Seymour Cray (1925-1996)

"father of supercomputing"



CRAY1: Vectorregisters (1976)

160 Mflop, 80 MHz, 8 MByte RAM

CRAY2: (1984)

1Gflop, 120MHz, 2GByte RAM

History

Supercomputer

JUGENE

IBM Blue Gene

At FZ Jülich,

Germany

Opening Ceremony June 2008

Computational Science...

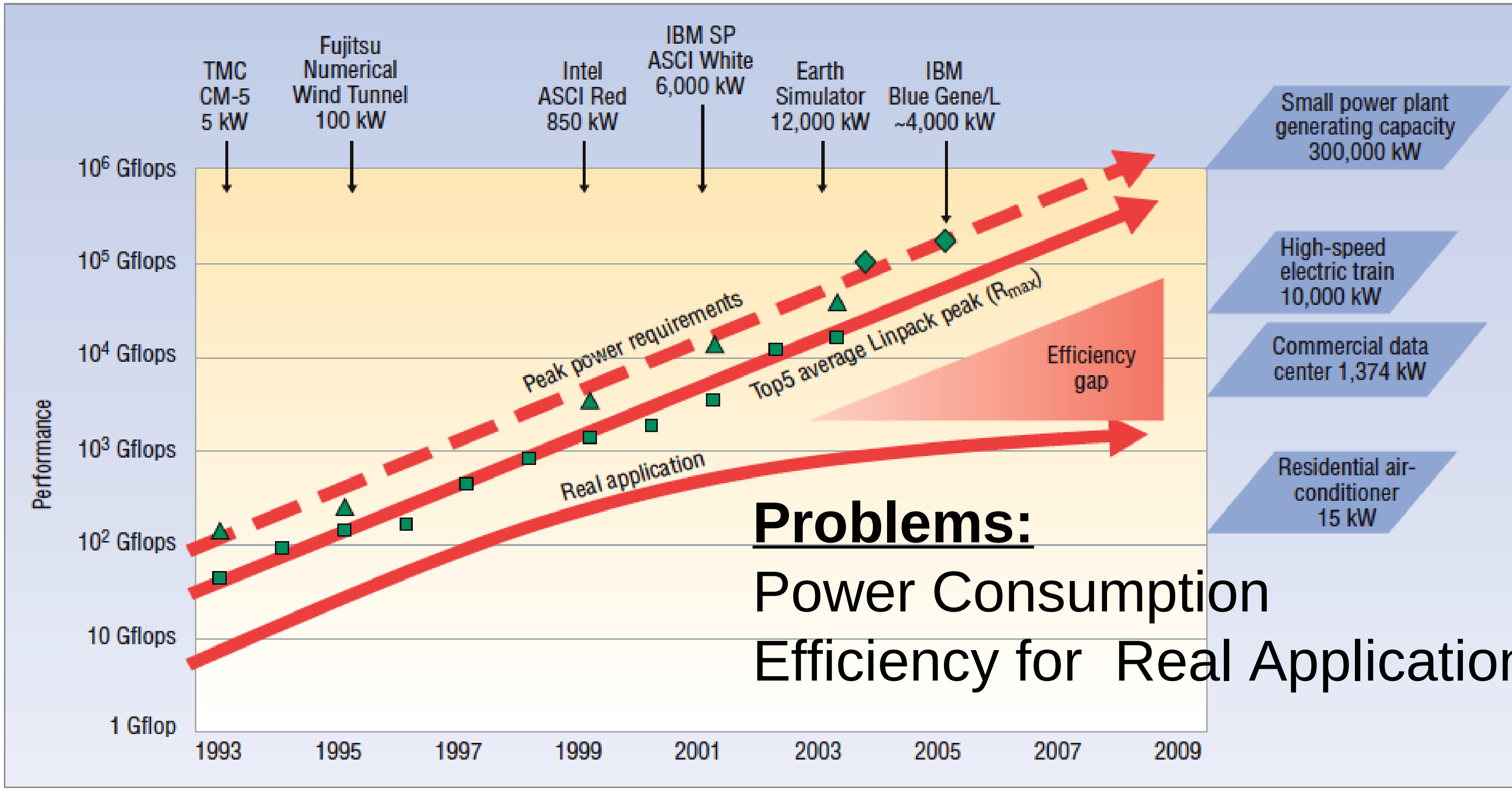
...after von Neumann...

Exaflop/s?

Petaflop/s

Teraflop/s

Gigaflop/s



Problems:

Power Consumption

Efficiency for Real Applications

Figure 1. Rising power requirements. Peak power consumption of the top supercomputers has steadily increased over the past 15 years.

Thanks to Horst Simon, LBNL/NERSC for this diagram.

GPU Computing

Special Hardware Accelerators

SPECIAL HARDWARE

CPUs

Central Processing Units

- General Purpose oriented
- 1-12 Cores
- Up to 4 pipes per core using Vector Units
- Fully Programmable, many languages available
- Very well studied
- Max. 125W per processor

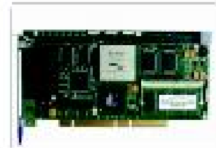
GPUs

Graphic Processing Units

- Graphics oriented
- 16-512 Cores
- Massively Parallel Architecture, specialized instructions for parallel processing
- Fully programmable, but limited languages
- Algorithms not fully explored
- Max. 400W per card

FPGAs

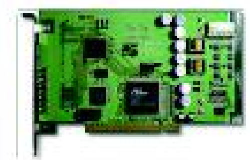
Field Programmable Gate Arrays



- Custom designs, best for processing streaming data
- Programmable Logic, Architecture is custom-built for the required application
- Requires extensive knowledge to program, development time is longer than CPUs and GPUs
- Application interface is custom built on each case
- Max. 60W per FPGA

ASICs

Application Specific Integrated Circuits



- Fully custom designs, built for a specific application
- Not flexible, cannot be changed once it is built
- Development is even more specialized than FPGAs
- Power consumption varies with the application, usually best performance per Watt

Slide: Guillermo Marcus

HITS

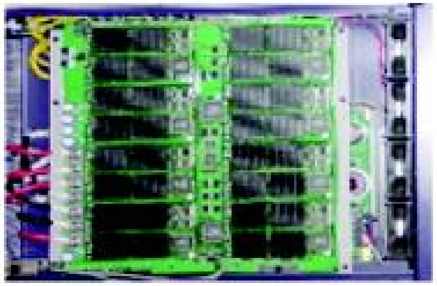
HARDWARE

GRAPE-6 Gravity/Coulomb Part

- G6 Chip: 0.25μ 2MGate ASIC, 6 Pipelines
- at 90MHz, 31Gflops/chip
- 48Tflops full system (March 2002)
- Plan up to 72Tflops full system (in 2002)
- Installed in Cambridge, Marseille, Drexel, Amsterdam, New York (AMNH), Mitaka (NAO), Tokyo, etc..

  New Jersey, Indiana, Heidelberg

GRAPE-6



1998, 120 Gflops

Developers: Junichiro Makino, Toshiyuki Fukushige, Hiroshi Daisaka, Eiichiro Kokubo, Masaki Koga, Makoto Taiji, Ken Namura

GRAPE-6: Massively-Parallel Special-Purpose Computer for Astrophysical Particle Simulations

Sales information

The Green500 List - November 2010

Listed below are the November 2010 The Green500's energy-efficient supercomputers ranked from 1 to 100.

http://www.green500.org

Green500 Rank	MFLOPS/W	Site*	Computer*	Total Power (kW)
1	1684.20	IBM Thomas J. Watson Research Center	NNSA/SC Blue Gene/Q Prototype	38.80
2+	1448.03	National Astronomical Observatory of Japan	GRAPE-DR accelerator Cluster, Infiniband	24.59
2	958.35	GSIC Center, Tokyo Institute of Technology	HP ProLiant SL390s G7 Xeon 6C X5670, Nvidia GPU, Linux/Windows	1243.80
3	933.06	NCSA	Hybrid Cluster Core i3 2.93Ghz Dual Core, NVIDIA C2050, Infiniband	36.00

GPU: NAOC laohu cluster Beijing, China

天河

"天河一号"超级计算机系统
TH-1 supercomputer

Landmark result of the important project "High Efficient Supercomputer and Grid Service Environment" supported by National 863 Program.

- Built by National University of Defense Technology, with the cooperation of National Supercomputer Center in Tianjin (NSCC -TJ) and Inspur (Beijing) Electronic Information Industry Co., Ltd.

Host system of NSCC-TJ, installed in Tianjin Binhai New Area.

A backbone node of the national grid of China.

Heidelberg

Kepler GPU cluster

VolkswagenStiftung

Kepler GPU cluster

12 nodes = 12 x 16 = 192 CPU cores (@ 2 GHz)

12 x 64 GB = 768 GB RAM CPU memory

12 GPUs K20m = 12 x 2496 ~ 30k GPU threads

12 x 4.8 GB ~ 57 GB GPU device memory

4 x Xilinx Virtex-6 FPGA (ML 605)

since beg. 2013 operated.

Top 10 List November 2010

From www.top500.org - list of fastest supercomputers in the world... ... last year Nov. 2010:

Rank	Site	Computer
1	National Supercomputing Center in Tianjin China	**Tianhe-1A** - NUDT TH MPP, X5670 2.93Ghz 6C, NVIDIA GPU, FT-1000 8C NUDT ***GPU***
2	DOE/SC/Oak Ridge National Laboratory United States	**Jaguar** - Cray XT5-HE Opteron 6-core 2.6 GHz Cray Inc.
3	National Supercomputing Centre in Shenzhen (NSCS) China	**Nebulae** - Dawning TC3600 Blade, Intel X5650, NVidia Tesla C2050 GPU Dawning ***GPU***
4	GSIC Center, Tokyo Institute of Technology Japan	**TSUBAME 2.0** - HP ProLiant SL390s G7 Xeon 6C X5670, Nvidia GPU, Linux/Windows NEC/HP ***GPU***
5	DOE/SC/LBNL/NERSC United States	**Hopper** - Cray XE6 12-core 2.1 GHz Cray Inc.
6	Commissariat a l'Energie Atomique (CEA) France FR	**Tera-100** - Bull bullx super-node S6010/S6030 Bull SA
7	DOE/NNSA/LANL United States	**Roadrunner** - BladeCenter QS22/LS21 Cluster, PowerXCell 8i 3.2 Ghz / Opteron DC 1.8 GHz, Voltaire Infiniband IBM
8	National Institute for Computational Sciences/University of Tennessee United States	**Kraken XT5** - Cray XT5-HE Opteron 6-core 2.6 GHz Cray Inc.
9	Forschungszentrum Juelich (FZJ) Germany	**JUGENE** - Blue Gene/P Solution IBM
10	DOE/NNSA/LANL/SNL United States	**Cielo** - Cray XE6 8-core 2.4 GHz Cray Inc.

China Grabs Supercomputing Leadership Spot in Latest Ranking of World's Top 500 Supercomputers

Thu, 2010-11-11 22:42

MANNHEIM, Germany; BERKELEY, Calif.; and KNOXVILLE, Tenn.—The 36th edition of the closely watched TOP500 list of the world's most powerful supercomputers confirms the rumored takeover of the top spot by the Chinese Tianhe-1A system at the National Supercomputer Center in Tianjin, achieving a performance level of 2.57 petaflop/s (quadrillions of calculations per second).

NCSA director: GPU is future of supercomputing

by Brooke Crothers

A A Font size Print E-mail Share 6 comments

Tweet 99 Share 25 2 Digg

The director of the National Center for Supercomputing Applications has seen the future of supercomputing and it can be summed up in three letters: GPU.

Thom Dunning, who directs the NCSA and the Institute for Advanced Computing Applications and Technologies at the famed supercomputing facilities on the campus of University of Illinois at Urbana-Champaign, says high-performance computing will begin to move toward graphics processing units or GPUs. Not coincidentally, **this is exactly what China has done to achieve the world's fastest speeds with its "Tianhe-1A"** supercomputer. That computer combines about 7,000 Nvidia GPUs with 14,000 Intel CPUs: the only hybrid CPU-GPU system in the world of that scale.

Thom Dunning directs the Institute for Advanced Computing Applications and Technologies and the NCSA.

"What we're really seeing in the efforts in China as well as the ones we have in the U.S. is that GPUs are what the future will look like," said Dunning in a phone interview Thursday. "What we're seeing is the beginning of something that's going to be happening all over the world."

NCSA already has a small CPU-GPU hybrid system. "It's something we have been working on for a number of years. We have a CPU-GPU cluster for the NCSA academic community. Made up of Intel CPUs and Nvidia GPUs. A 50 teraflop machine," he said. (Note that **Oak Ridge National Laboratories is also installing a hybrid system now**.)

Intel MIC Hardware

INSPUR, NAOC - 2013.XI.26

icpc … "-mmic" … 61 x 4 = 244 x 1.1 GHz omp cores !!!

Full fp64 !!!

Intel MIC Hardware

Intel® Xeon Phi™ Coprocessor Family Reference Table

SKU #	Form Factor, Thermal	Peak Double Precision	Max # of Cores	Clock Speed (GHz)	GDDR5 Memory Speeds (GT/s)	Peak Memory BW	Memory Capacity (GB)	Total Cache (MB)	Board TDP (Watts)	Process
SE10P (special edition)	PCIe Card, Passively Cooled	1073 GF	61	1.1	5.5	352	8	30.5	300	22nm
SE10X (special edition)	PCIe Card, No Thermal Solution	1073 GF	61	1.1	5.5	352	8	30.5	300	
5110P	PCIe Card, Passively Cooled	1011 GF	60	1.053	5.0	320	8	30	225	
3100 Series	PCIe Card, Actively Cooled	>1 TF	Disclosed at 3100 series launch (H1'13)		5.0	240	6	28.5	300	
	PCIe Card, Passively Cooled	> 1 TF			5.0	240	6	28.5	300	

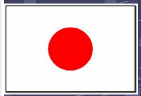
PCIe Card, Actively Cooled

PCIe Card, Passively Cooled

Current Generation:
Knights Landing
14nm

Top 10 List 2011 ------ 2012

1	Sequoia - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom
2	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect
3	Mira - BlueGene/Q, Power BQC 16C 1.60GHz, Custom
4	SuperMUC - iDataPlex DX360M4, Xeon E5-2680 8C 2.70GHz, Infiniband FDR
5	Tianhe-1A - NUDT YH MPP, Xeon X5670 6C 2.93 GHz, NVIDIA 2050
6	Jaguar - Cray XK6, Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA 2090
7	Fermi - BlueGene/Q, Power BQC 16C 1.60GHz, Custom
8	JuQUEEN - BlueGene/Q, Power BQC 16C 1.60GHz, Custom
9	Curie thin nodes - Bullx B510, Xeon E5-2680 8C 2.700GHz, Infiniband QDR
10	Nebulae - Dawning TC3600 Blade System, Xeon X5650 6C 2.66GHz, Infiniband QDR, NVIDIA 2050



Rank	Site	System
1	National University of Defense Technology China ***Xeonφ***	**Tianhe-2 (MilkyWay-2)** - TH-IVB-FEP Cluste Intel Xeon E5-2692 12C 2.200GHz, TH Expre Intel Xeon Phi 31S1P NUDT
2	DOE/SC/Oak Ridge National Laboratory United States ***GPU***	**Titan** - Cray XK7 , Opteron 6274 16C 2.200G Cray Gemini interconnect, NVIDIA K20x Cray Inc.
3	DOE/NNSA/LLNL United States	**Sequoia** - BlueGene/Q, Power BQC 16C 1.6 GHz, Custom IBM
4	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu
5	DOE/SC/Argonne National Laboratory United States	**Mira** - BlueGene/Q, Power BQC 16C 1.60GH Custom IBM
6	Texas Advanced Computing Center/Univ. of Texas United States ***Xeonφ***	**Stampede** - PowerEdge C8220, Xeon E5-26 8C 2.700GHz, Infiniband FDR, Intel Xeon Phi SE10P Dell
7	Forschungszentrum Juelich (FZJ) Germany	**JUQUEEN** - BlueGene/Q, Power BQC 16C 1.600GHz, Custom Interconnect IBM
8	DOE/NNSA/LLNL United States	**Vulcan** - BlueGene/Q, Power BQC 16C 1.600GHz, Custom Interconnect IBM
9	Leibniz Rechenzentrum Germany	**SuperMUC** - iDataPlex DX360M4, Xeon E5-2 8C 2.70GHz, Infiniband FDR

Nr. 1,2 Supercomputer from China: 96/33 Pflop/s Linpack
Wuxi/Guangzhou/Tianjin National Supercomputing Center
Taihu 10 mill. cores

Tianhe-2 (MilkyWay-2) - TH-IV
E5-2692 12C 2.200GHz, TH Ex
31S1P

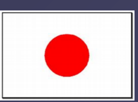
32000 Intel Xeon 12 core,
48000 Intel Phi Accelerators 57 Core

Test of Taihu planned;
But:
Local cluster with new GPUs at NAOC gives much more resources.

Top 10 List June 2016


USA

USA

USA

USA

Swiss

Saudi-A.

Chinese Processor

Xeonϕ

GPU

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	National Supercomputing Center in Wuxi China	**Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRCPC	10,649,600	93,014.6	125,435.9	15,371
2	National Super Computer Center in Guangzhou China	**Tianhe-2 (MilkyWay-2)** - TH-IVB-FEP Cluster, Intel Xeo E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P NUDT	3,120,000	33,862.7	54,902.4	17,808
3	DOE/SC/Oak Ridge National Laboratory United States	**Titan** - Cray XK7 , Opteron 6274 16C 2.200GHz, Cray Gemini intercor NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209
4	DOE/NNSA/LLNL United States	**Sequoia** - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1,572,864	17,173.2	20,132.7	7,890
5	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu	705,024	10,510.0	11,280.4	12,660
5	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu	705,024	10,510.0	11,280.4	12,660
6	DOE/SC/Argonne National Laboratory United States	**Mira** - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	786,432	8,586.6	10,066.3	3,945
7	DOE/NNSA/LANL/SNL United States	**Trinity** - Cray XC40, Xeon E5-2698v3 16C 2.3GHz, Aries interconnect Cray Inc.	301,056	8,100.9	11,078.9	
8	Swiss National Supercomputing Centre (CSCS) Switzerland	**Piz Daint** - Cray XC30, Xeon E5-2670 8C 2.600GHz, Aries interconnect , NVIDIA K20x Cray Inc.	115,984	6,271.0	7,788.9	2,325
9	HLRS - Höchstleistungsrechenzentrum Stuttgart Germany	**Hazel Hen** - Cray XC40, Xeon E5-2680v3 12C 2.5GHz, Aries interconnect Cray Inc.	185,088	5,640.2	7,403.5	
10	King Abdullah University of Science and Technology Saudi Arabia	**Shaheen II** - Cray XC40, Xeon E5-2698v3 16C 2.3GHz, Aries interconnect Cray Inc.	196,608	5,537.0	7,235.2	2,834

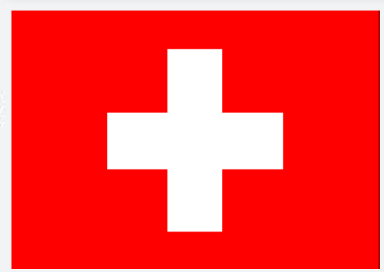
TOP500 List Refreshed, US Edged Out of Third Place ■ ■ ■ ■

TOP500 Team | June 19, 2017 00:22 CEST

FRANKFURT, Germany; BERKELEY, Calif.; and KNOXVILLE, Tenn.— The 49th edition of the TOP500 list was released today in conjunction with the opening session of the ISC High Performance conference, which is taking place this week in Frankfurt, Germany. The list ranks the world's most powerful supercomputers based on the Linpack benchmark and is released twice per year.

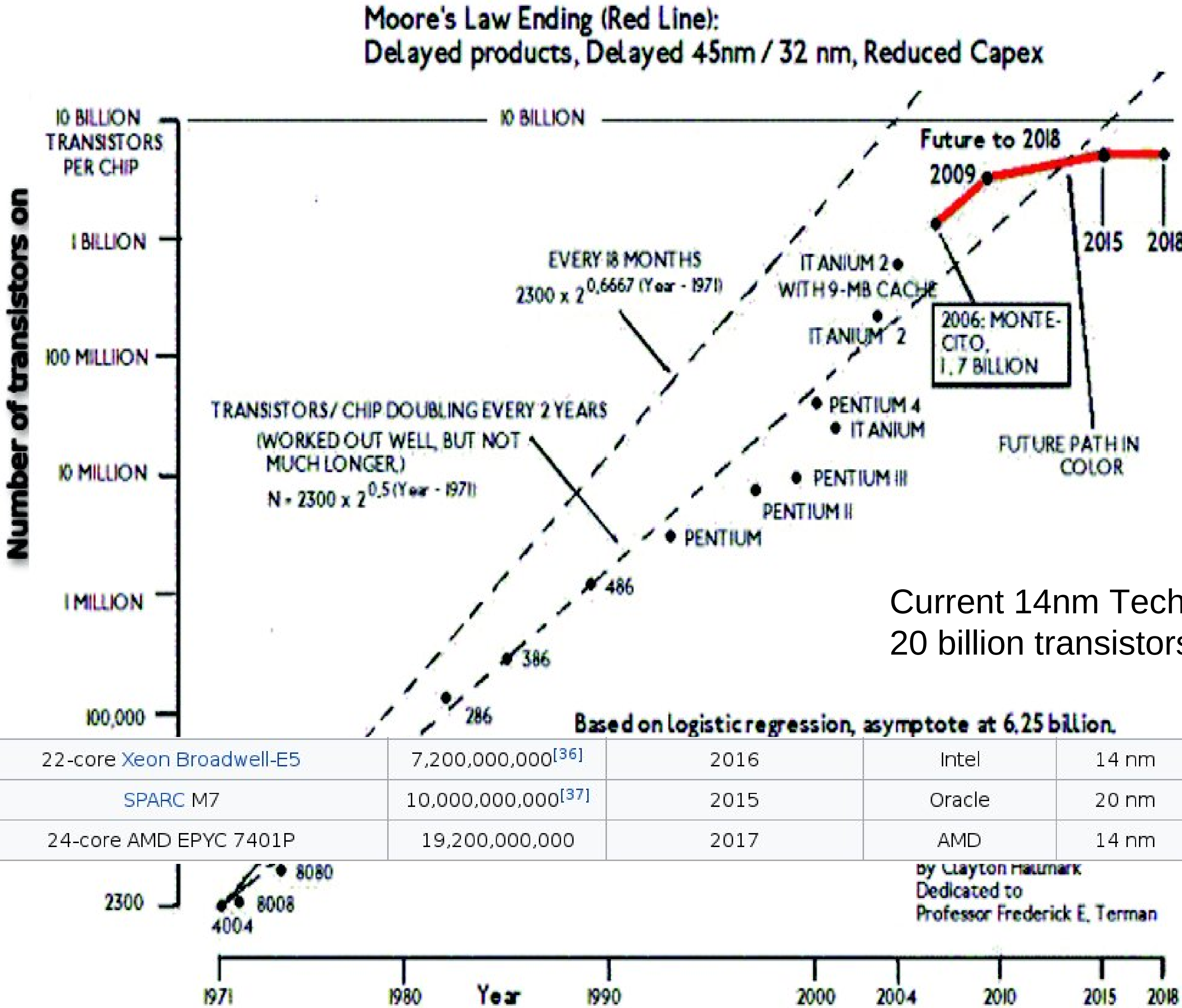
Read more

System
Sunway TaihuLight - Sunway MPP, Sunway SW26010 260C 1.45(Sunway , **NRCPC** National Supercomputing Center in Wuxi **China**
Tianhe-2 (MilkyWay-2) - TH-IVB-FEP Cluster, Intel Xeon E5-26(2.200GHz, TH Express-2, Intel Xeon Phi 31S1P , **NUDT** National Super Computer Center in Guangzhou **China**
Piz Daint - Cray XC50, Xeon E5-2690v3 12C 2.6GHz, Aries interc(NVIDIA Tesla P100 , **Cray Inc.** Swiss National Supercomputing Centre (CSCS) **Switzerland**



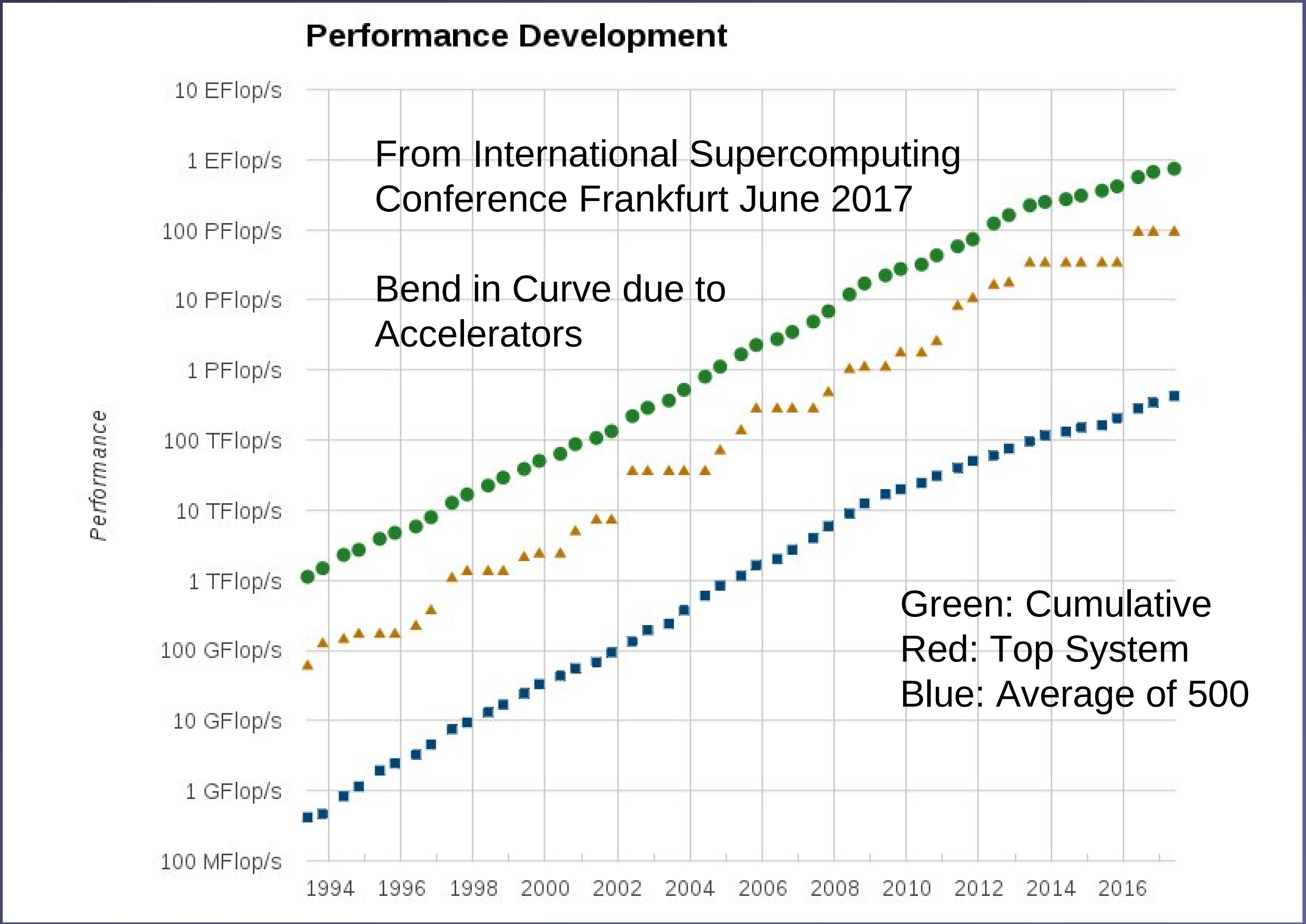
■ ■ ■ ■

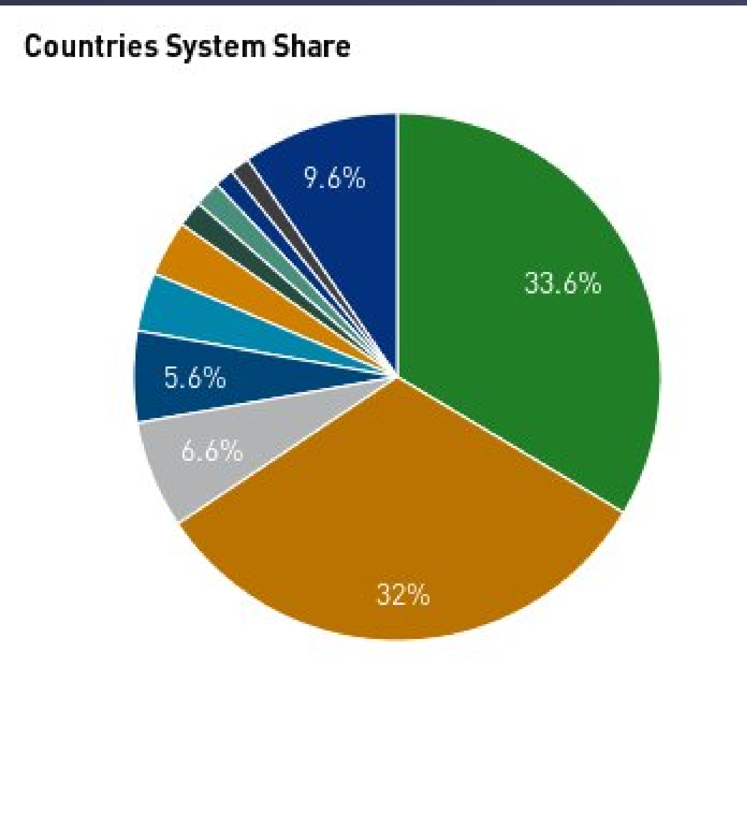
By Switzerland

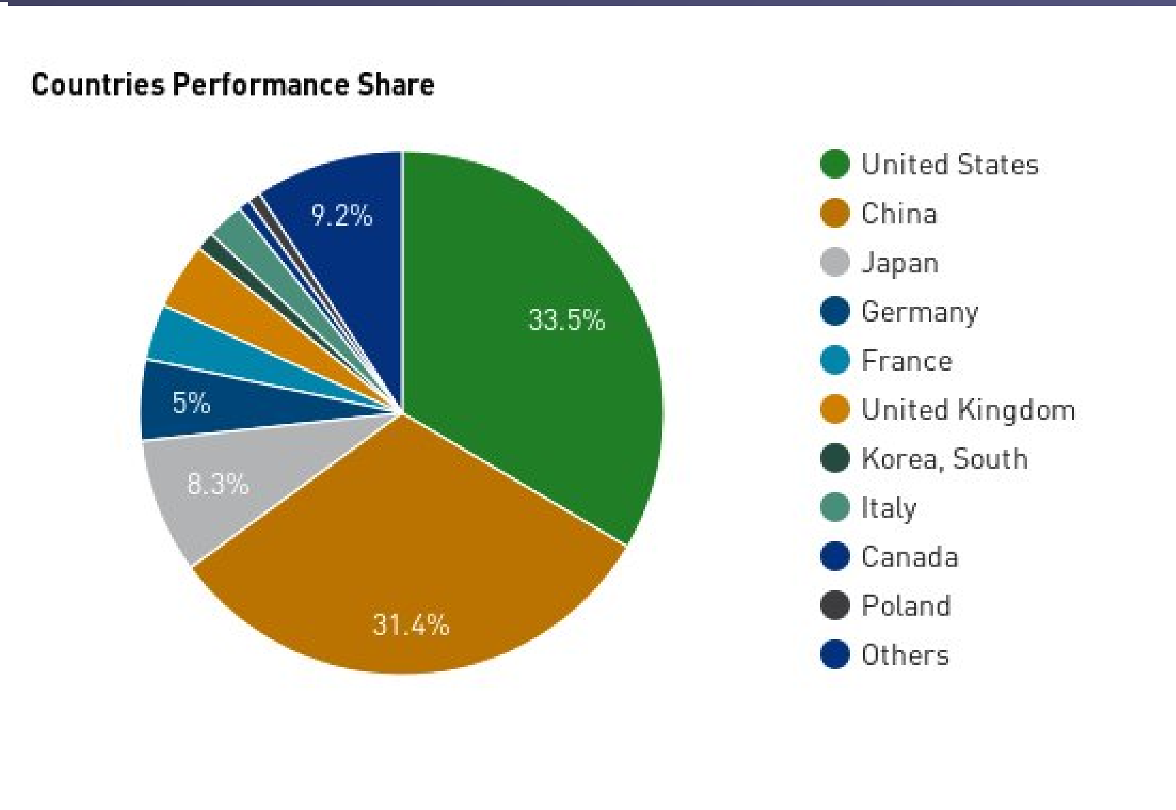


Current 14nm Technology
20 billion transistors

22-core Xeon Broadwell-E5	7,200,000,000[36]	2016	Intel	14 nm	456 mm²
SPARC M7	10,000,000,000[37]	2015	Oracle	20 nm	
24-core AMD EPYC 7401P	19,200,000,000	2017	AMD	14 nm	195 mm²


Performance Development
10 EFlop/s
1 EFlop/s
100 PFlop/s
10 PFlop/s
1 PFlop/s
100 TFlop/s
10 TFlop/s
1 TFlop/s
100 GFlop/s
10 GFlop/s
1 GFlop/s
100 MFlop/s
Performance
1994
1996
1998
2000
2002
2004
2006
2008
2010
2012
2014
2016
From International Supercomputing
Conference Frankfurt June 2017
Bend in Curve due to
Accelerators
Green: Cumulative
Red: Top System
Blue: Average of 500


Countries System Share
9.6%
33.6%
5.6%
6.6%
32%

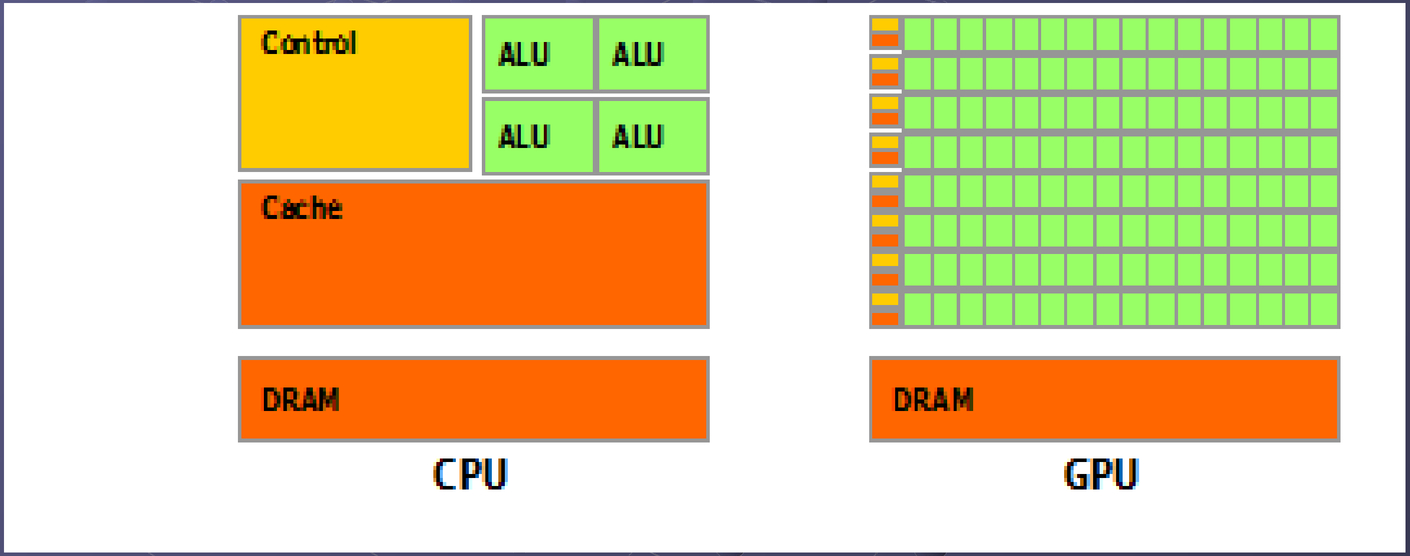

Countries Performance Share
9.2%
33.5%
5%
8.3%
31.4%
United States
China
Japan
Germany
France
United Kingdom
Korea, South
Italy
Canada
Poland
Others

More on GPU

Graphics Processors (GPU) as General Purpose Supercomputers (GPGPU)

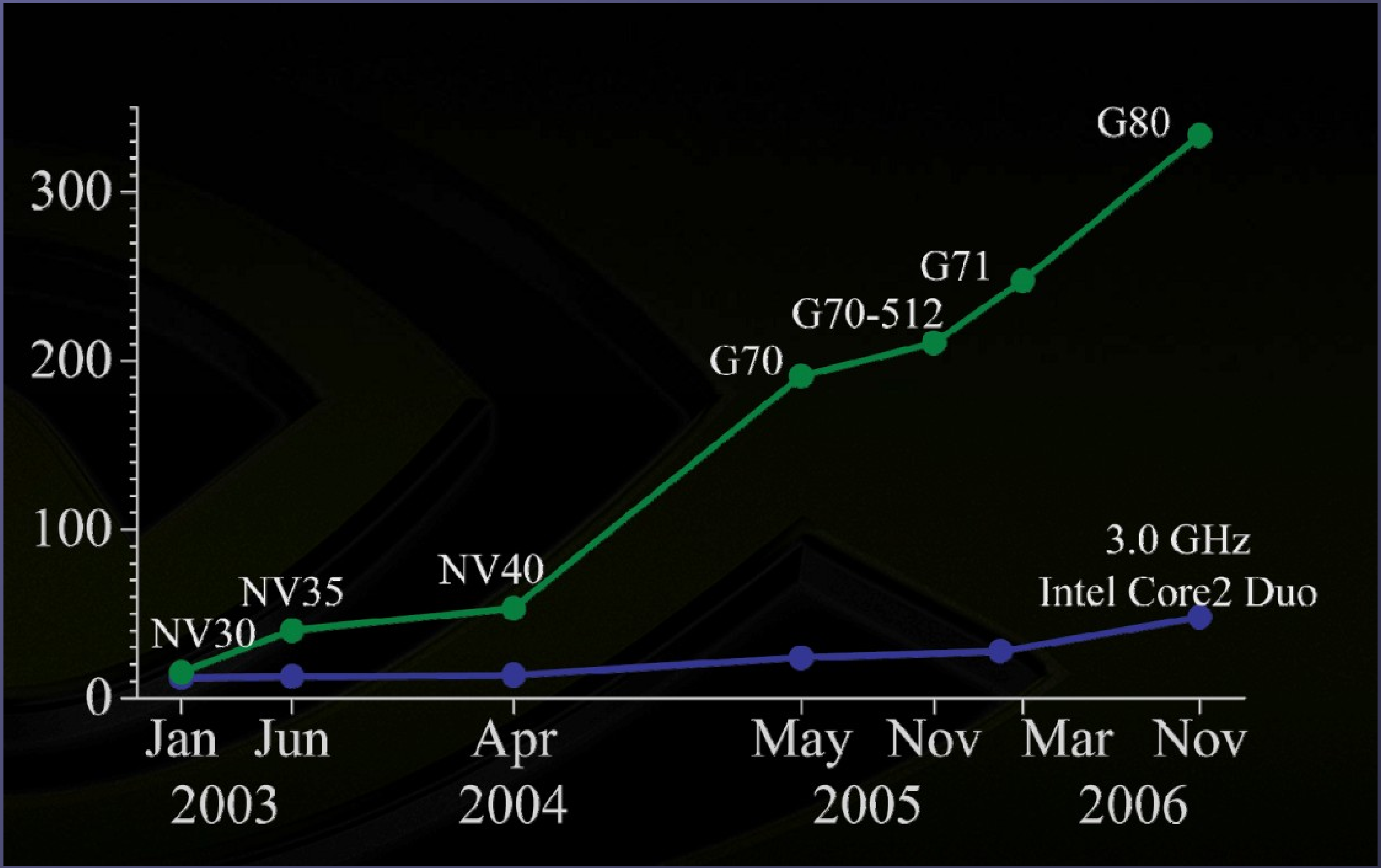
2008…
GeForce 9800 GTX, 128 Stream Proc., 512 MB
GeForce 9800 GX2, 256 Stream Proc., 1 GB
GeForce 9800 GT, 64 Stream Proc., 512 MB
[…]
2009: Tesla ~200 Proc., 4GB
2010: Fermi ~400 Proc., 4GB
2013: Kepler K20, ~2500 Procs., 6GB
2016: Kepler K80, ~5000 Procs.
2017/18: Pascal, Volta > 5000 Procs.

CPU and GPU; from CUDA NVIDIA Developer Zone at
http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html

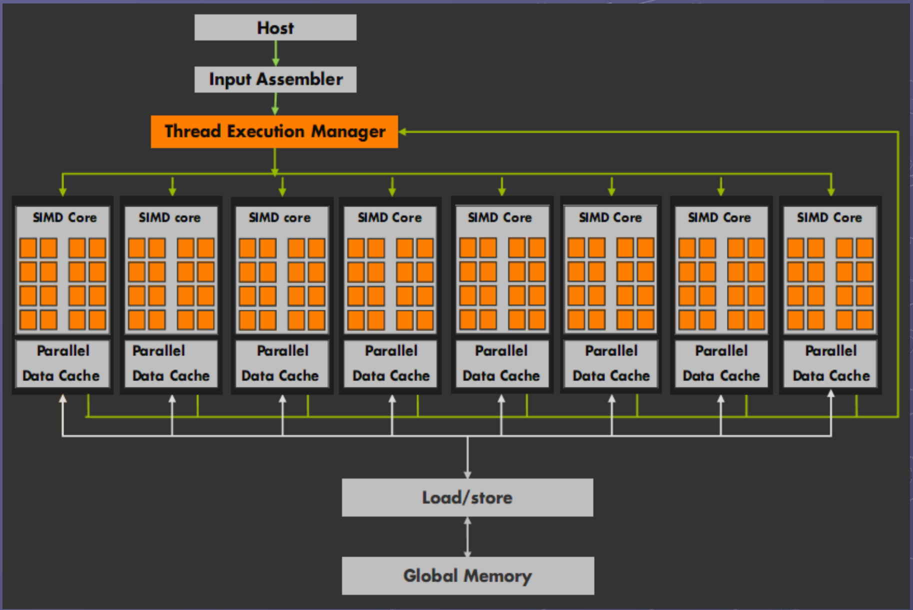


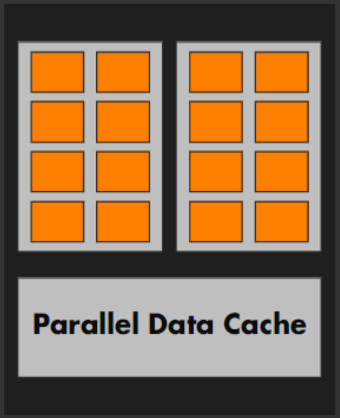
“The GPU devotes more transistors to computing”
“favours data parallel operations”

CPU vs. GPU speedup timeline



Hardware around 2006





Each core

- 8 functional units
- SIMD 16/32 “warp”
- 8-10 stage pipeline
- Thread scheduler
- 128-512 threads/core
- 16 KB shared memory

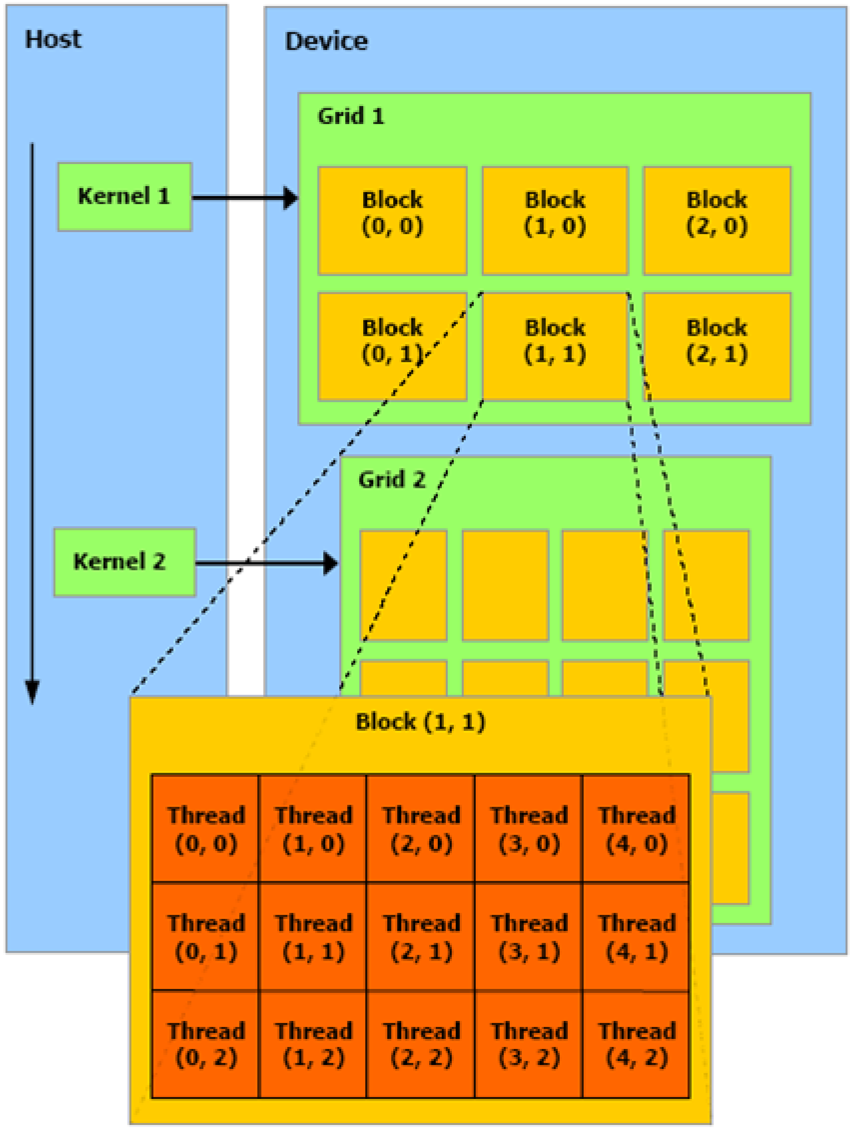
Total #threads/chip

16 * 512 = 8K

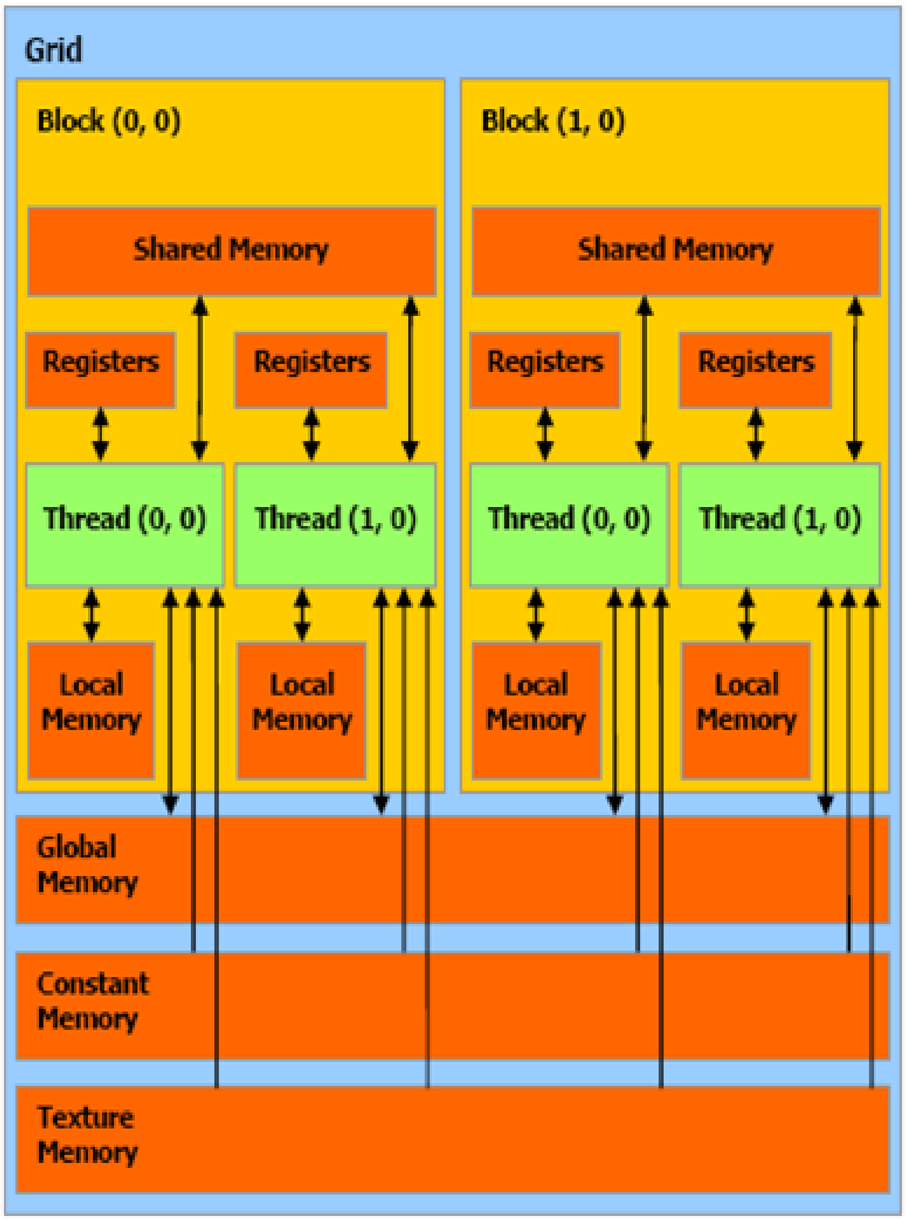
GeForce 8800 GTX:

575 MHz * 128 processors * 2 flop/inst * 2 inst/clock = 333 Gflops

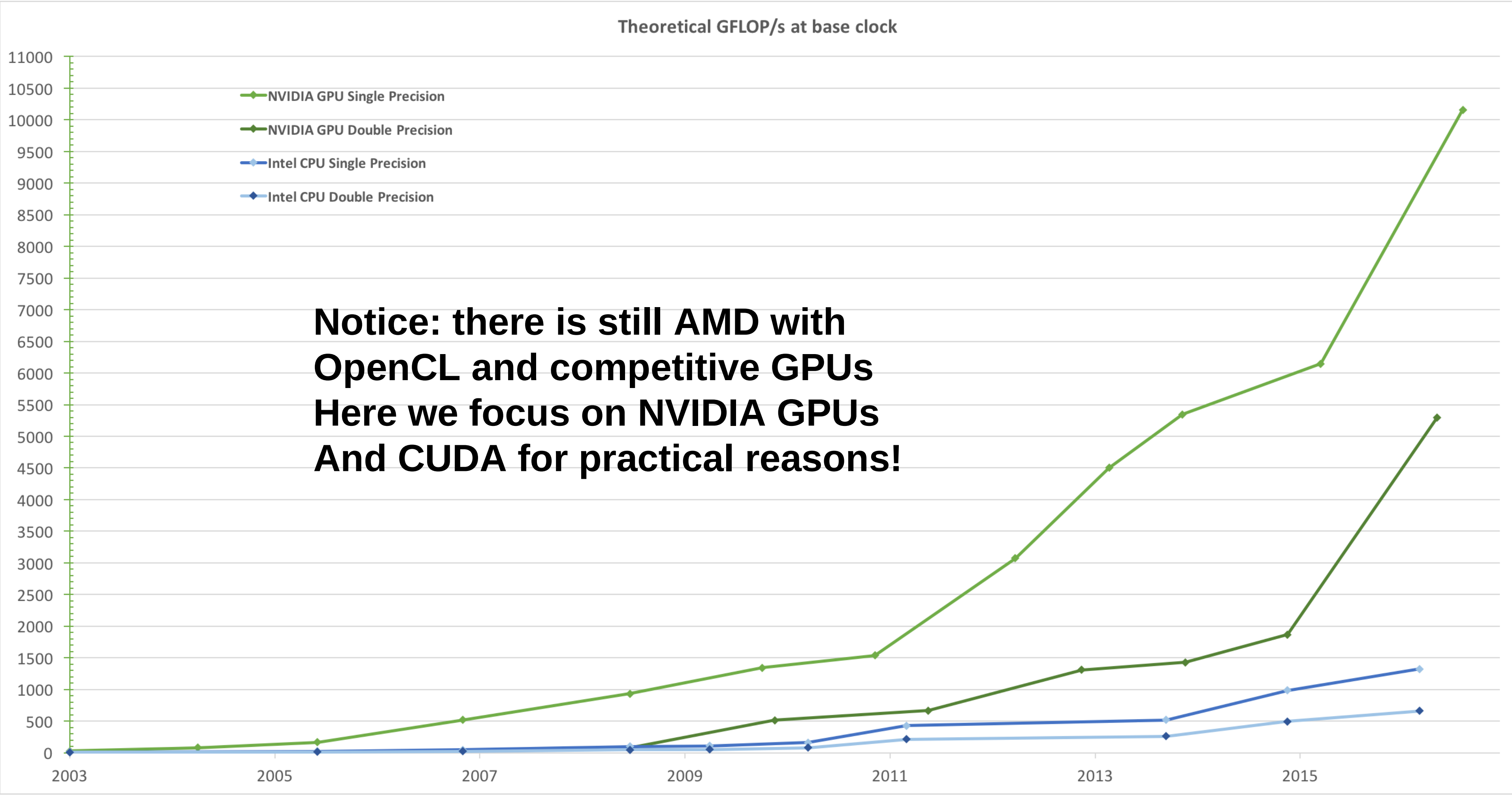
GPU Structure From: http://geco.mines.edu/tesla/cuda_tutorial_mio/



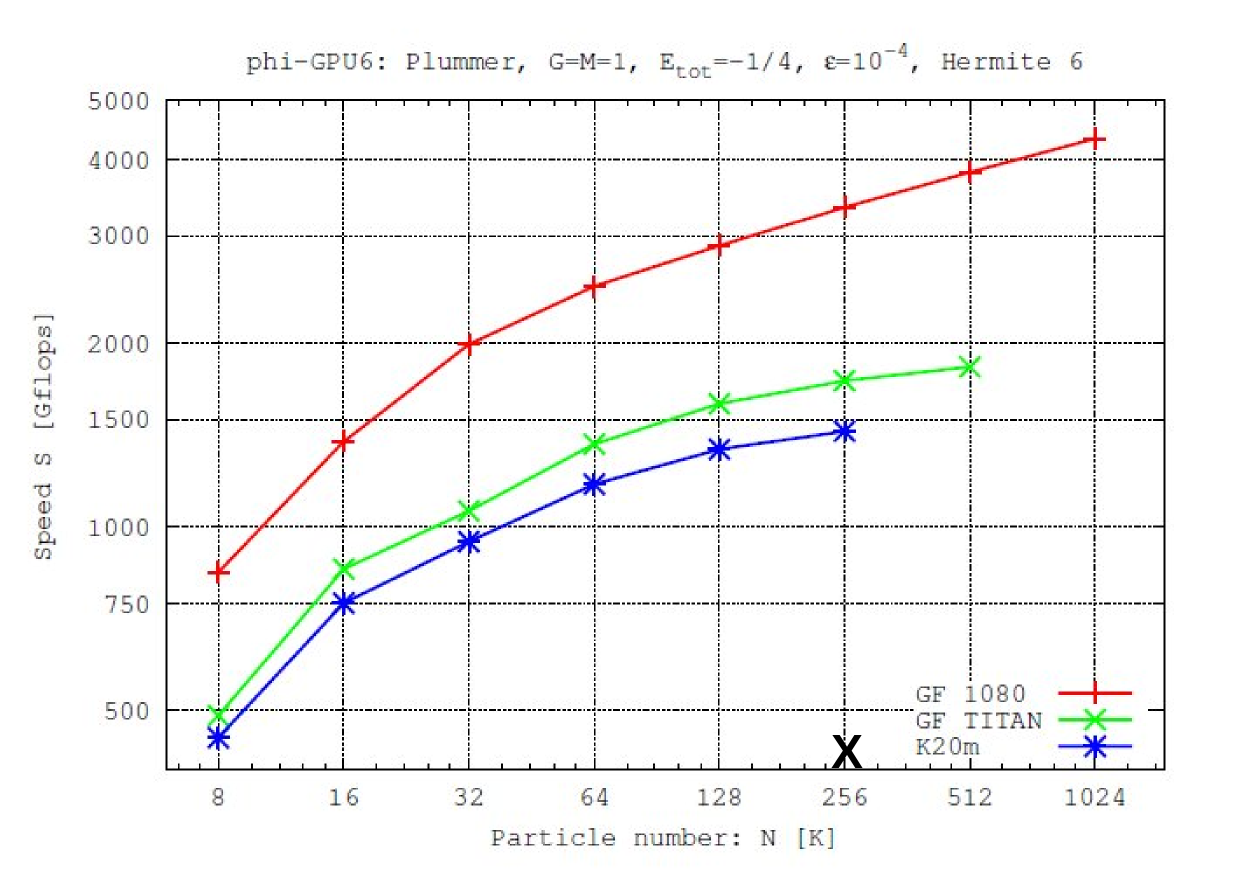
The host issues a succession of kernel invocations to the device. Each kernel is executed as a batch of threads organized as a grid of thread blocks



<u>Floating Point Operations per Second for CPU and GPU:</u>
From NVIDIA CUDA Developer Zone at:
http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html



Kepler, Pascal Scaling, it works...



Pascal GF1080

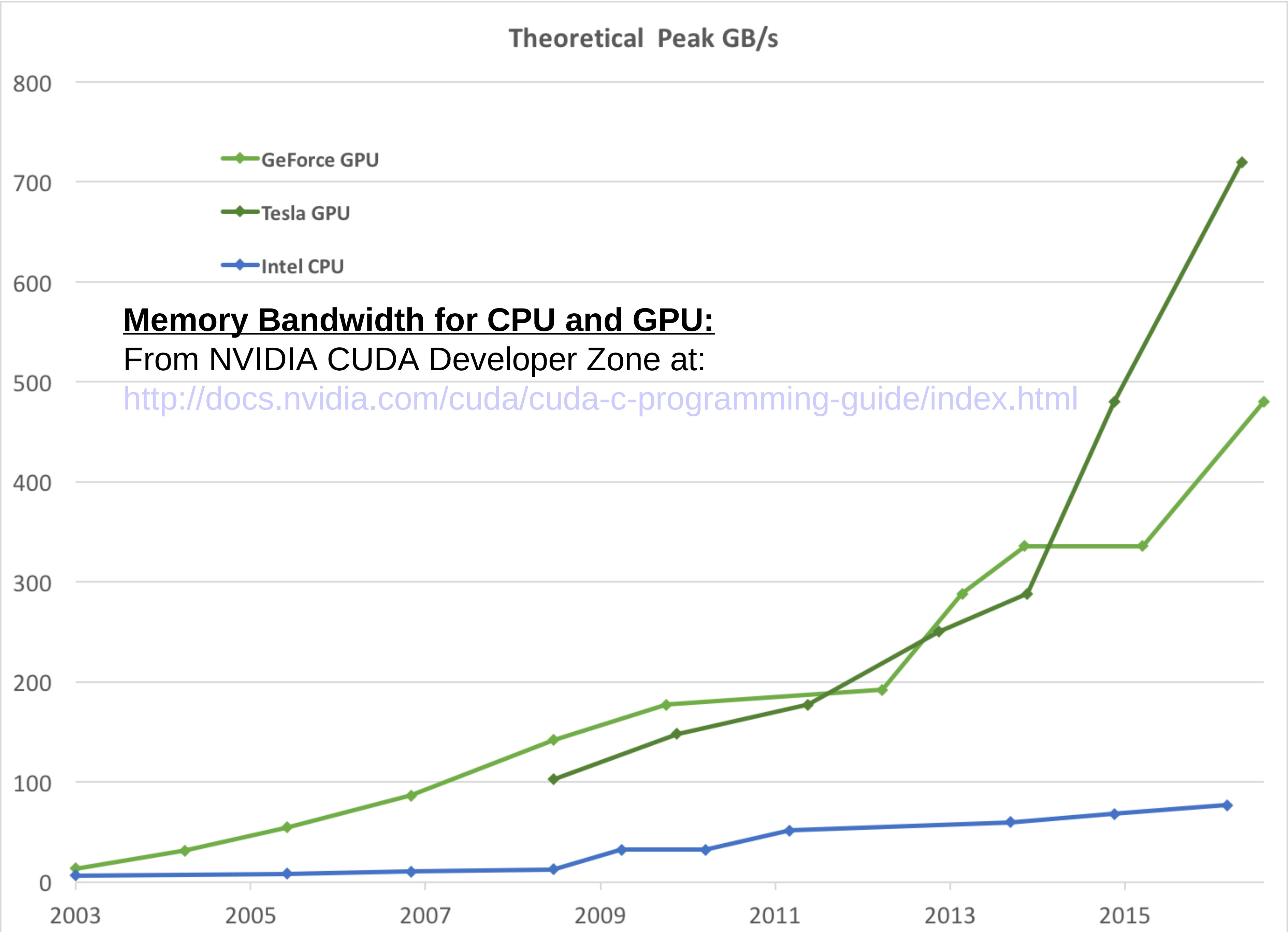
TITAN (Kepler)

Kepler K20m

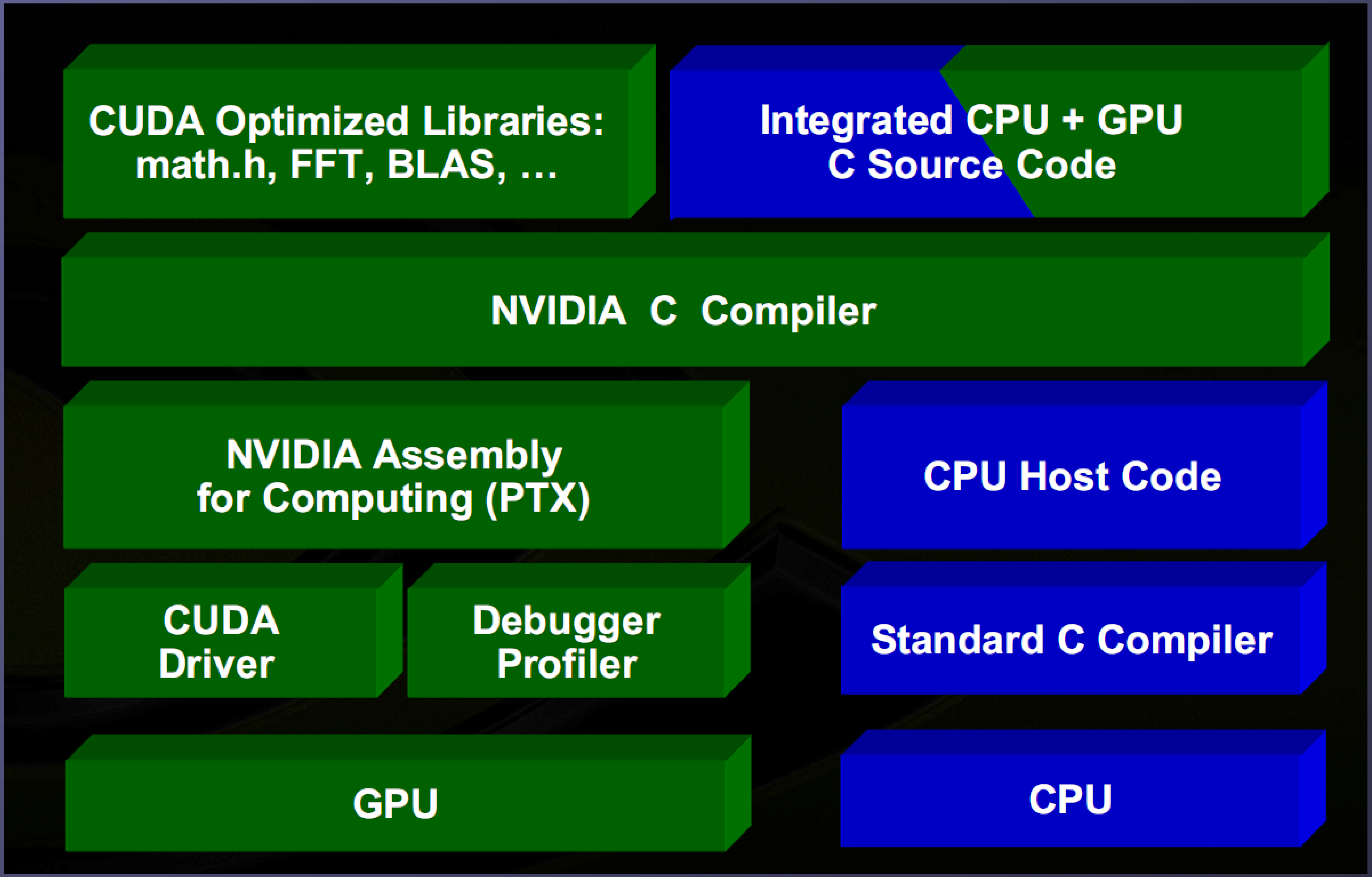
Spurzem, Berczik, et al., 2013, LNCS Supercomputing, 2013, pp. 13-25, Springer publisher.

Fig. 4. Here we report a preliminary result from a benchmark test of our code on one Kepler K20 card; we compare with the performance on Fermi C2050 (used in the Mole-8.5 cluster), and the oldest Tesla C1060 GPU (used in the laohu cluster of 2009) - the latter is used as a normalization reference. We plot the speed ratio of our usual benchmarking simulation used in the previous figures, as a function of particle number. From this we see the sustained performance of a Kepler K20 would be about 1.4 - 1.5 Tflop/s.

X = first GPU of laohu 2010


Theoretical Peak GB/s
800
700
600
500
400
300
200
100
0
2003
2005
2007
2009
2011
2013
2015
GeForce GPU
Tesla GPU
Intel CPU
Memory Bandwidth for CPU and GPU:
From NVIDIA CUDA Developer Zone at:
http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html

CUDA


CUDA Optimized Libraries:
math.h, FFT, BLAS, …
Integrated CPU + GPU
C Source Code
NVIDIA C Compiler
NVIDIA Assembly
for Computing (PTX)
CPU Host Code
CUDA
Driver
Debugger
Profiler
Standard C Compiler
GPU
CPU

Simple CUDA example

CPU C program

```
void addMatrix(float *a, float *b,
               float *c, int N)
{
  int i, j, index;
  for (i = 0; i < N; i++) {
    for (j = 0; j < N; j++) {
      index = i + j * N;
      c[index]=a[index] + b[index];
    }
  }
}

void main()
{
  .....
  addMatrix(a, b, c, N);
}
```

CUDA C program

```
__global__ void addMatrix(float *a,float *b,
                          float *c, int N)
{
  int i=blockIdx.x*blockDim.x+threadIdx.x;
  int j=blockIdx.y*blockDim.y+threadIdx.y;
  int index = i + j * N;
  if ( i < N && j < N)
    c[index]= a[index] + b[index];
}

void main()
{
  ..... // allocate & transfer data to GPU
  dim3 dimBlk (blocksize, blocksize);
  dim3 dimGrd (N/dimBlk.x, N/dimBlk.y);
  addMatrix<<<dimGrd,dimBlk>>>(a, b, c,N);
}
```

GPU Computing Applications

Libraries and Middleware

Source: http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html

cuDNN TensorRT	cuFFT cuBLAS cuRAND cuSPARSE	CULA MAGMA	Thrust NPP	VSIPL SVM OpenCurrent	PhysX OptiX iRay	MATLAB Mathematica

Programming Languages

C	C++	Fortran	Java Python Wrappers	DirectCompute	Directives (e.g. OpenACC)

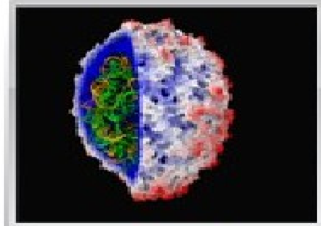
CUDA-Enabled NVIDIA GPUs

Volta Architecture (compute capabilities 7.x)				Tesla V Series
Pascal Architecture (compute capabilities 6.x)		GeForce 1000 Series	Quadro P Series	Tesla P Series
Maxwell Architecture (compute capabilities 5.x)	Tegra X1	GeForce 900 Series	Quadro M Series	Tesla M Series
Kepler Architecture (compute capabilities 3.x)	Tegra K1	GeForce 700 Series GeForce 600 Series	Quadro K Series	Tesla K Series

Speedups using GPU vs. CPU

146X

Interactive visualization of volumetric white matter connectivity[1]

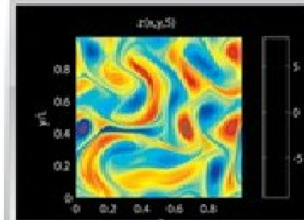


36X

Ionic placement for molecular dynamics simulation on GPU[2]

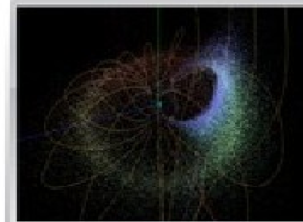
18X

Transcoding HD video stream to H.264 for portable video[3]



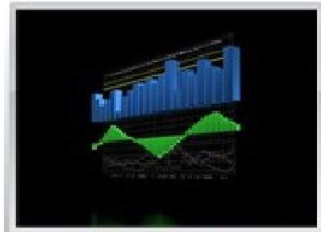
17X

Simulation in Matlab using .mex file CUDA function[4]



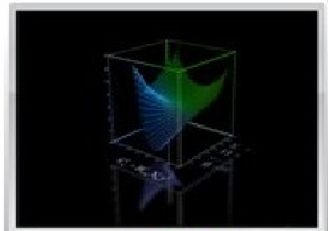
100X

Astrophysics N-body simulation[5]



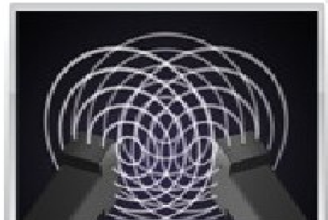
149X

Financial simulation of LIBOR model with swaptions[6]



47X

GLAME@lab: M-script API for linear Algebra operations on GPU[7]

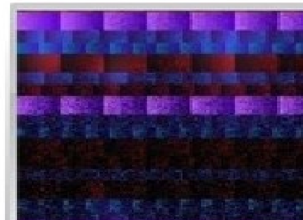


20X

Ultrasound medical imaging for cancer diagnostics[8]

24X

Highly optimized object oriented molecular dynamics[9]



30X

Cmatch exact string matching - find similar proteins & gene sequences[10]

Towards Peta-Scale Green Computation

— applications of the GPU supercomputers in CAS

http://www.nvidia.com/gtc2010-content

GPU TECHNOLOGY CONFERENCE

GTC 2010 | Sept 20-23, 2010
San Jose Convention Center, San Jose, California
Watch the Keynote Recordings

- Algorithms & Numerical Techniques
- Astronomy & Astrophysics
- Audio Processing
- Cloud Computing
- Computational Fluid Dynamics
- Computer Graphics
- Computer Vision
- Databases & Data Mining
- Digital Content Creation
- Embedded & Automotive
- Energy Exploration
- Film
- Finance
- General Interest
- GPU Accelerated Internet
- High Performance Computing
- Imaging
- Life Sciences
- Machine Learning & Artificial Intelligence
- Medical Imaging & Visualization
- Mobile & Tablet & Phone
- Molecular Dynamics
- Neuroscience
- Physics Simulation
- Programming Languages & Techniques
- Quantum Chemistry
- Ray Tracing
- Signal Processing
- Stereoscopic 3D
- Tools & Libraries
- Video Processing

Wei Ge
Xiaowei Wang
Inst. of Proc. Eng.

Yunquan Zhang
Inst. of Software

Rainer Spurzem
Nat. Astro. Obs. Chn.

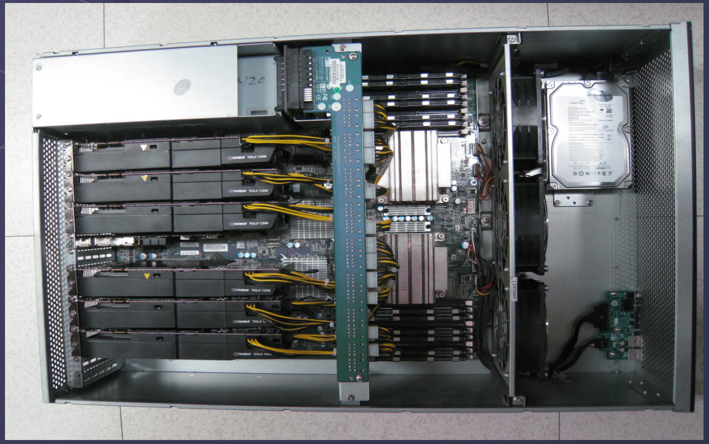
Long Wang
SC Center

Computer Physics - Astrophysics

Molecular Dynamics

Fermi-based GPU supercomputer IPE (2010.04.24)

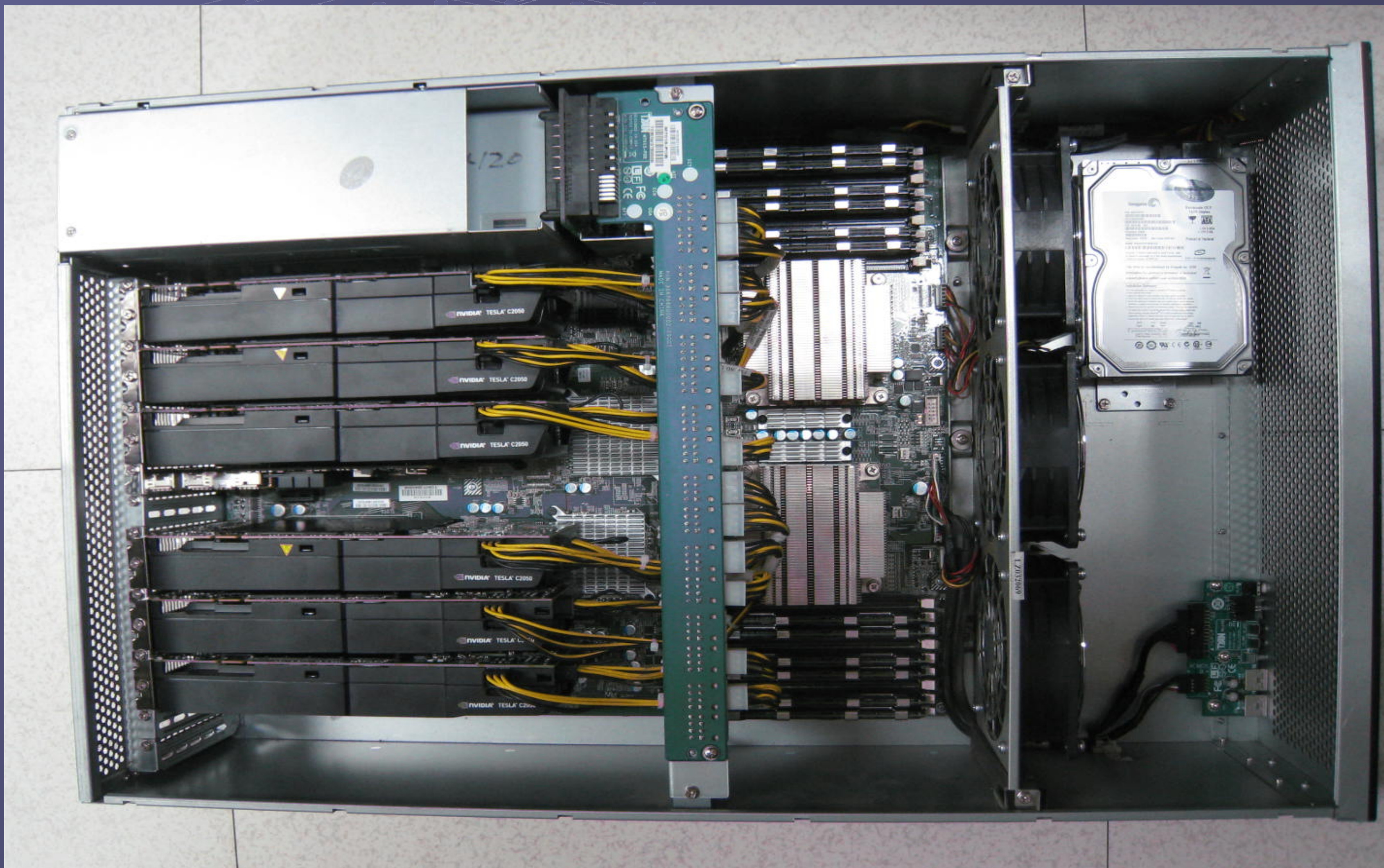
Rpeak SP：	2Pflops
Rpeak DP：	1Pflops
Linpack:	207.3T (Top500 19th)
Mflops/Watt:	431 (Green500 8th)
Total RAM：	17.2TB
Total VRAM：	6.6TB
Total HD：	360TB
Inst. Comm.：	H3C GE
Data Comm.：	Mellanox QDR IB
Occupied area：	150 sq.m.
Weight：	12.6 tons
Max Power：	600kW(computing) 200kW(cooling)
System：	CentOS 5.4, PBS
Monitor：	Ganglia, GPU monitor
Languages：	C, C++, CUDA 3.1 , OpenCL



中国科学院过程工程研究所
Institute Of Process Engineering, Chinese Academy Of Sciences

IPE CAS 372 node 6xC2050 cluster

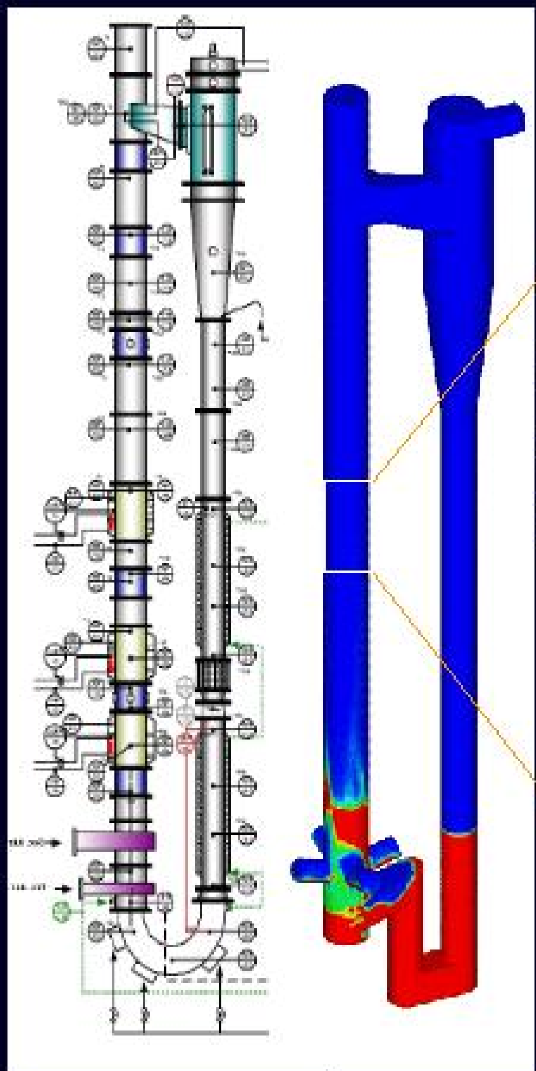
2232 GPU = 2.2 Pflops SP / 1.1 Pflops DP



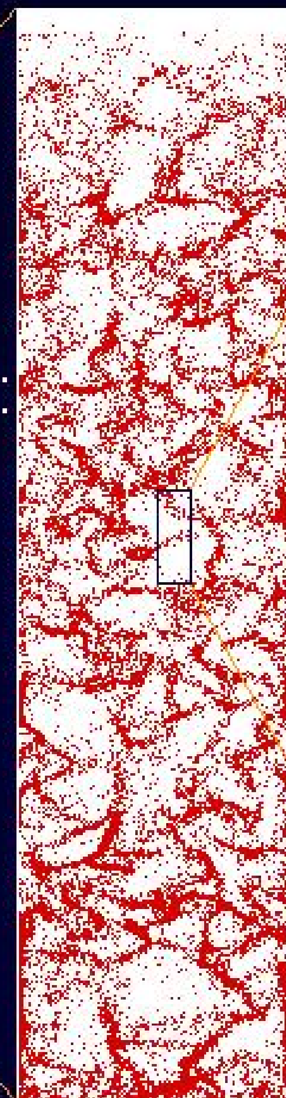
DNS of gas-solid flow： >20x speedup (1C1060/1E5430 core)

120K Particles + 400M pseudo-particles

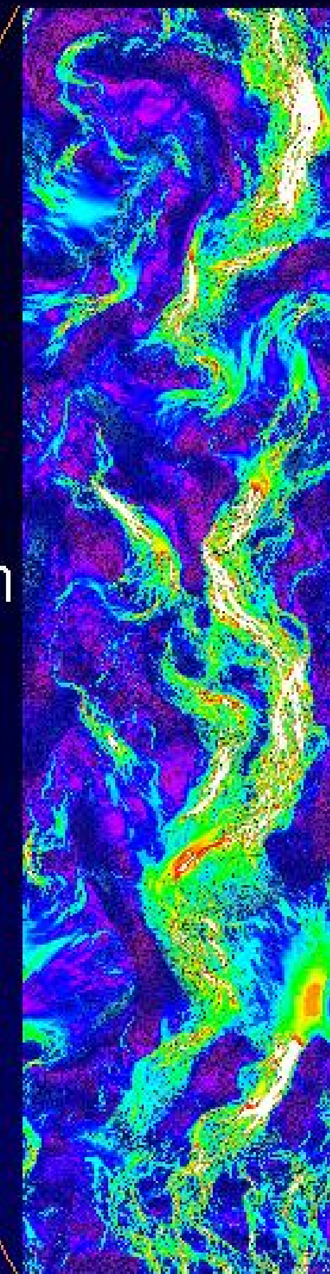
Reactor:
0.4*20m
3D



Section:
0.4*1m
2D



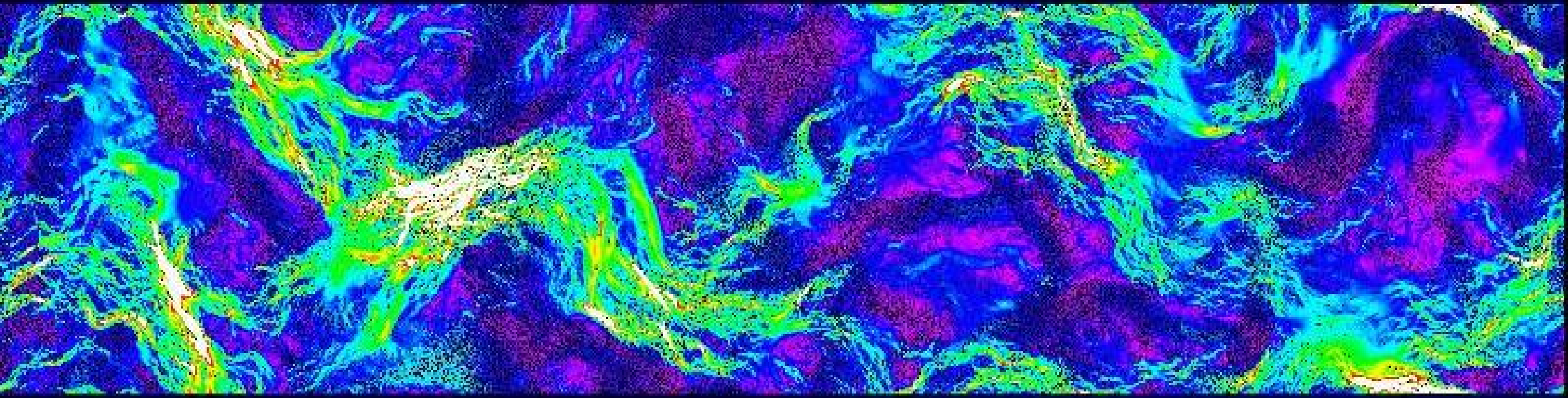
Cell:
2*10cm
2D



Animation Challenge:

9600x2400 → 1200x300 pixels

1000 → 17 frames



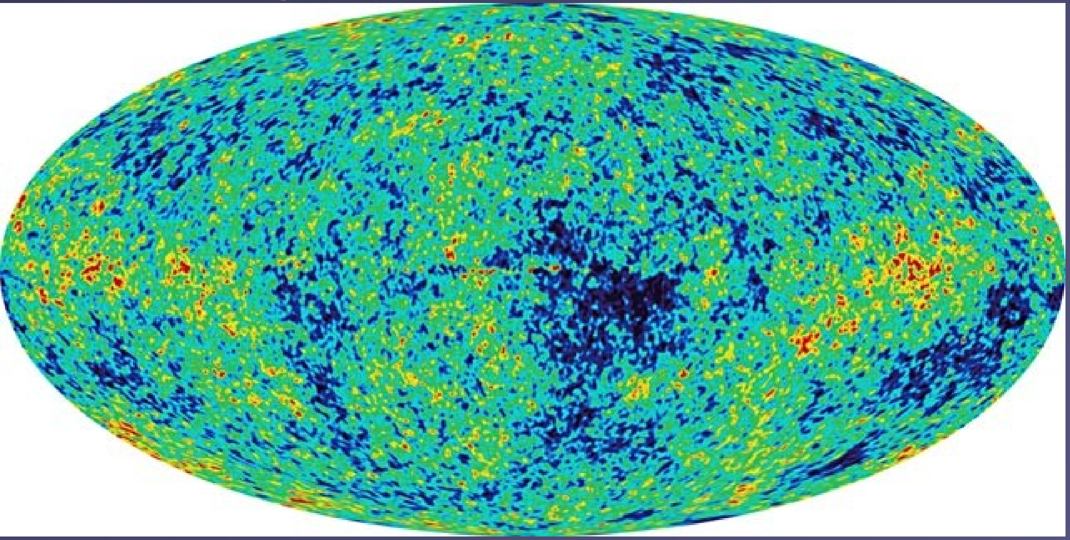
Computer Physics - Astrophysics

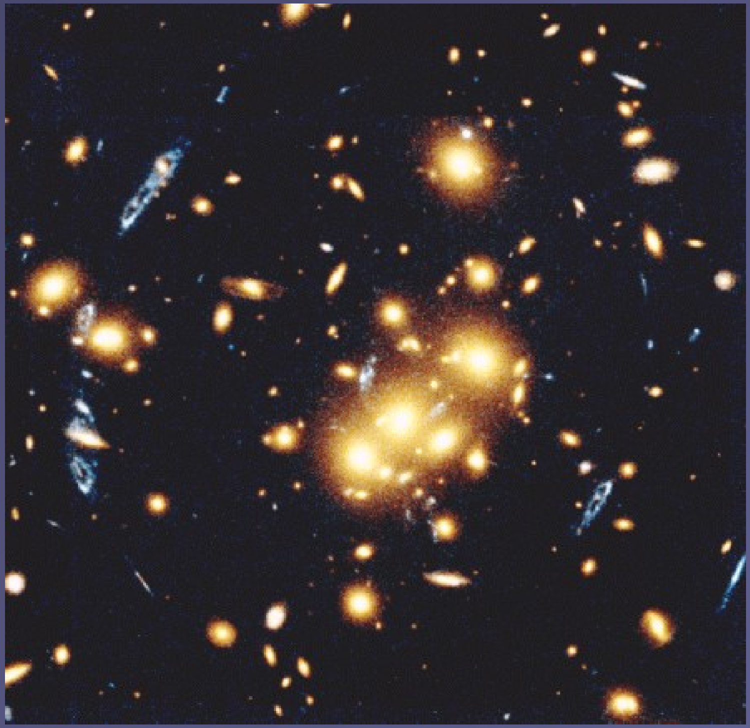
Cosmology

Computer Physics – Astrophysics

- Structure Formation in the Universe

In the year 100.000....

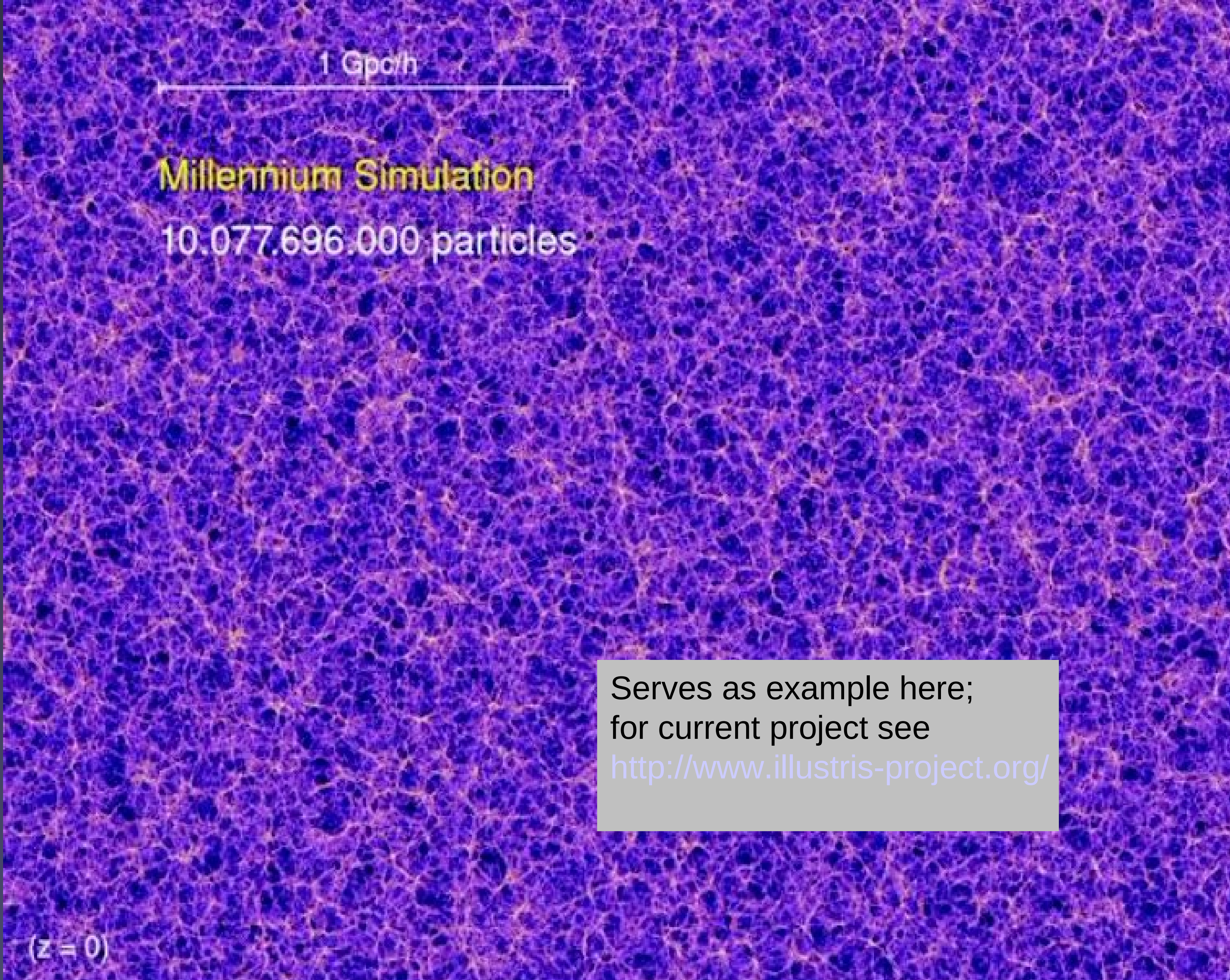




- Wilkinson Microwave Anisotropy Probe (WMAP)

(Cosmic Microwave Background)

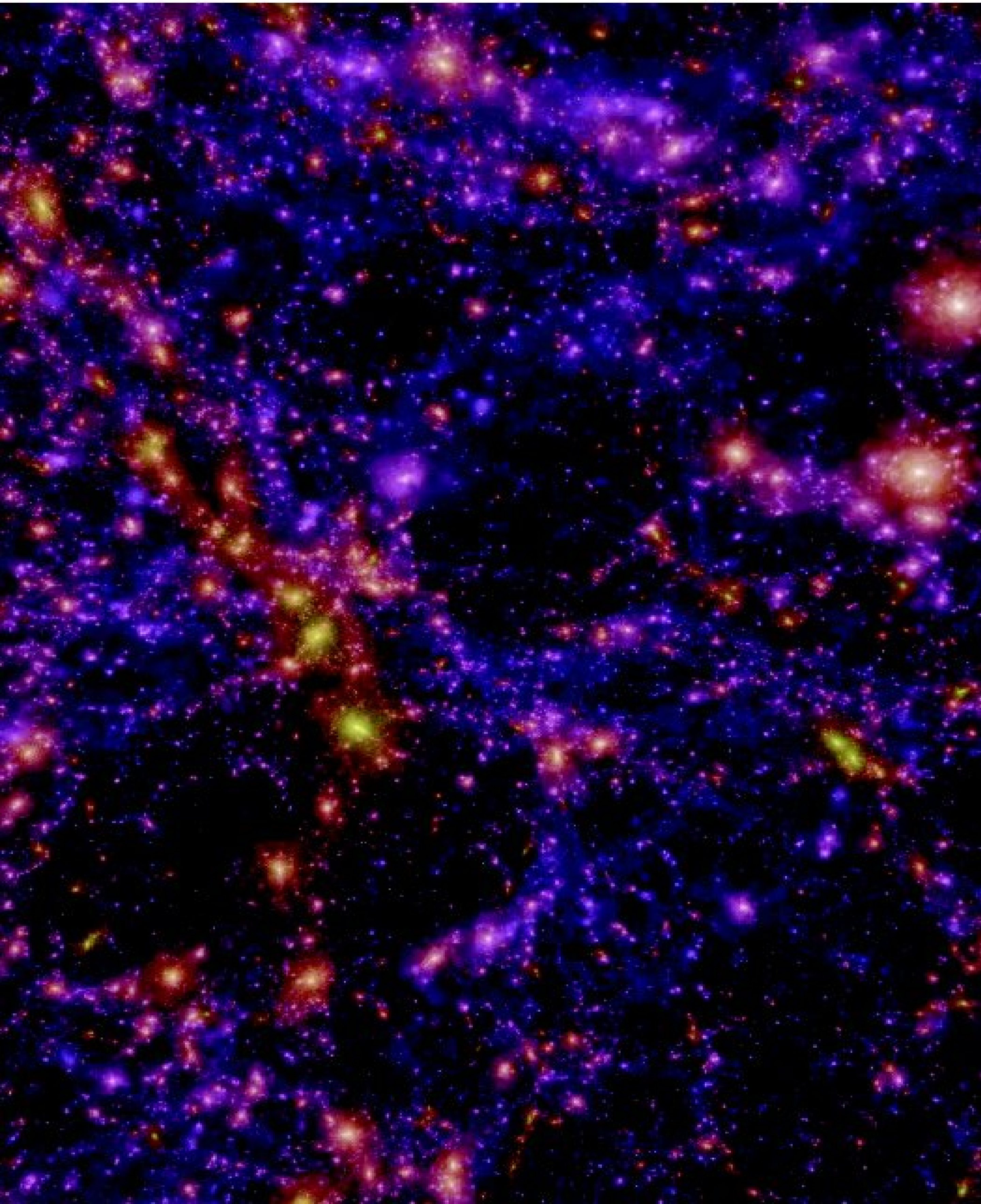
...and ``today´´

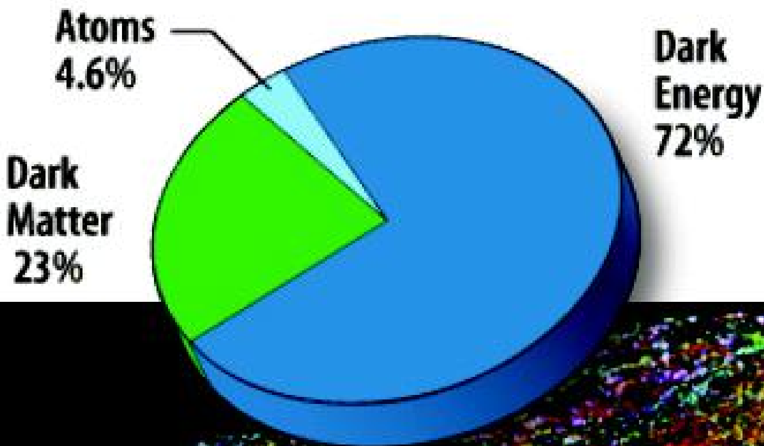


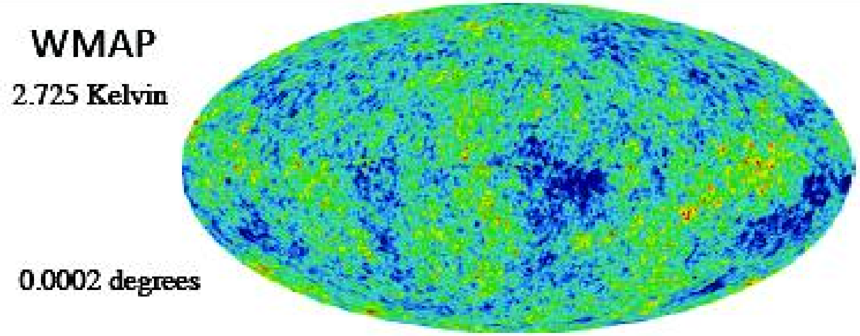
Serves as example here;
for current project see
http://www.illustris-project.org/

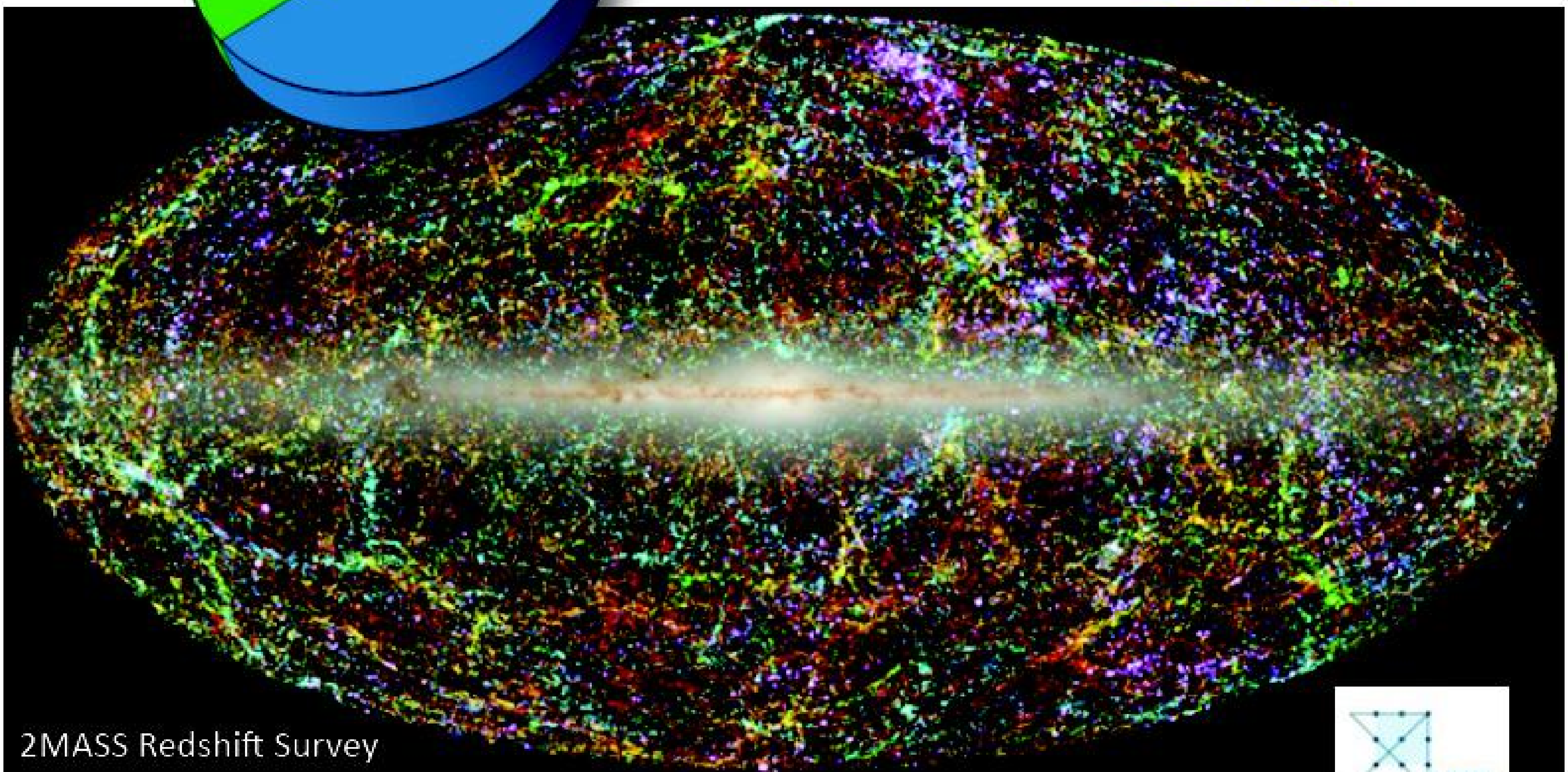
Millenium Simulaiton (Springel et al.)

Millenium Simulaiton (Springel et al.)



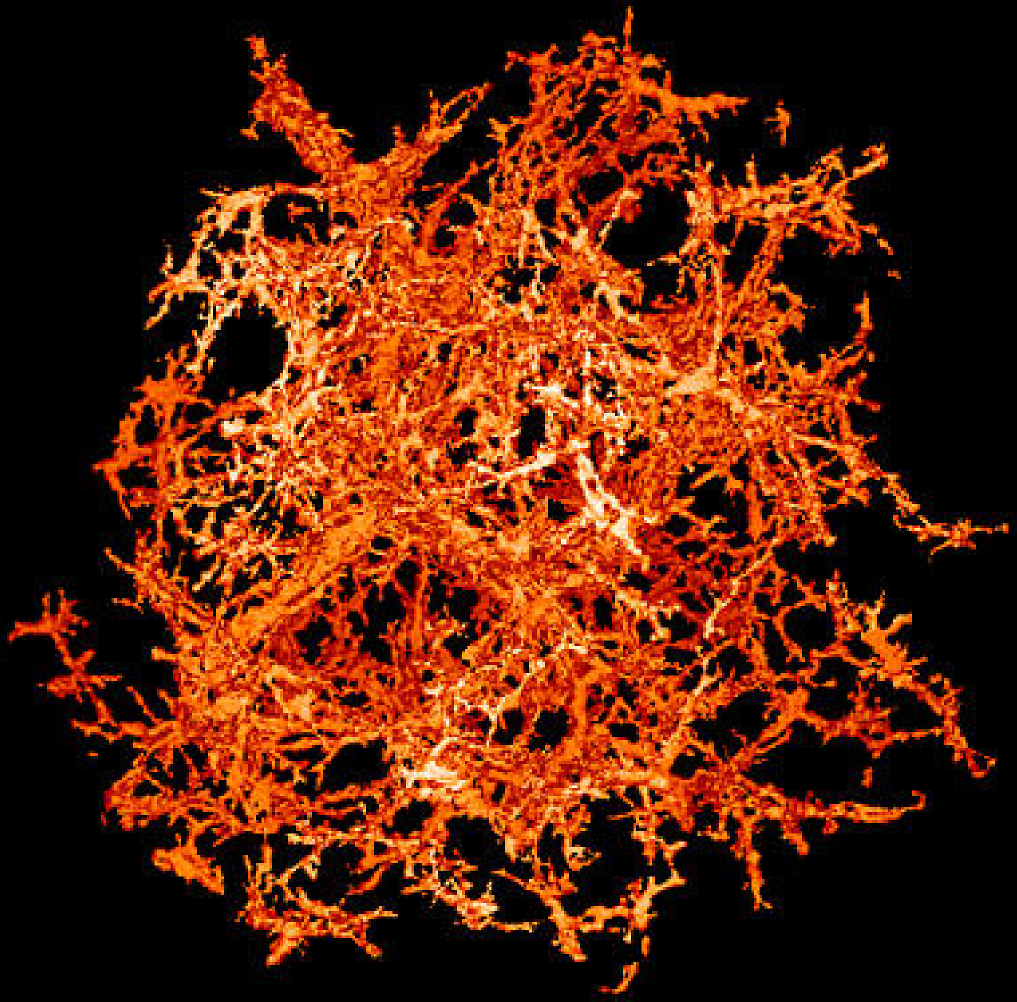






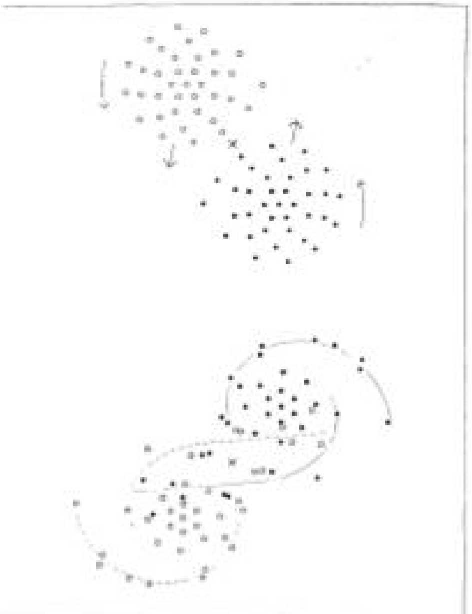
(Image: T.H. Jarrett (IPAC/SSC))

Challenges: Cosmology



Structure of Voids and Filaments in 3D

(From Virgo-Webpages)



Holmberg, 1937/1941

NGC 4038/NGC 4039

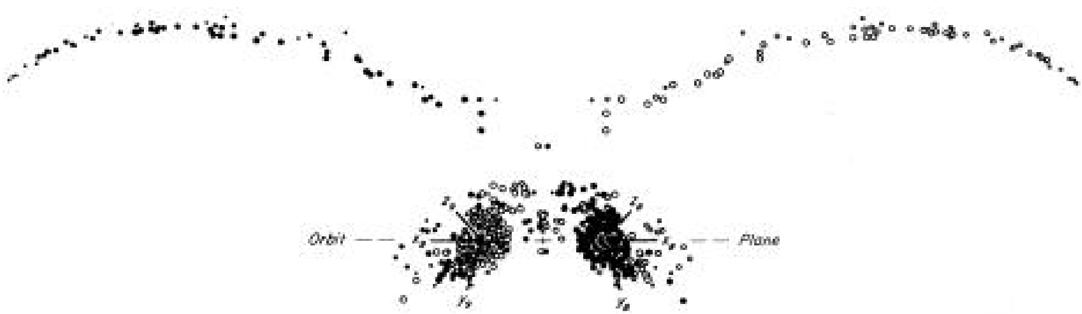


FIG. 23.—Symmetric model of NGC 4038/9. Here two identical disks of radius $0.75R_{min}$ suffered an $e \approx 0.5$ encounter with orbit angles $i_8 = i_9 = 60°$ and $\omega_8 = \omega_9 = -30°$ that appeared the same to both. The above all-inclusive views of the debris and remnants of these disks have been drawn exactly normal and edge-on to the orbit plane; the latter viewing direction is itself 30° from the line connecting the two pericenters. The viewing time is $t = 15$, or slightly past apocenter. The filled and open symbols again disclose the original loyalties of the various test particles.

Toomre & Toomre,1972, ApJ, 178, 623

HITS

Computer Physics - Astrophysics

Black Holes in Star Clusters

MPA Garching
Highlight March 2016

http://www.mpa-garching.mpg.de/328833/hl201603

http://kiaa.pku.edu.cn News...

HIGHLIGHT: MARCH 2016

The DRAGON globular cluster simulations: a million stars, black holes and gravitational waves

March 01, 2016

An international team of experts from Europe and China has performed the first simulations of globular clusters with a million stars on the high-performance GPU cluster of the Max Planck Computing and Data Facility. These – up to now - largest and most realistic simulations can not only reproduce observed properties of stars in globular clusters at unprecedented detail but also shed light into the dark world of black holes. The computer models produce high quality synthetic data comparable to Hubble Space Telescope observations. They also predict nuclear clusters of single and binary black holes. The recently detected gravitational wave signal might have originated from a binary black hole merger in the center of a globular cluster.

RGB image of a simulated globular cluster

© MPA

Globular clusters are truly enigmatic objects. They consist of hundreds of thousands luminous stars and their remnants, which are confined to a few tens of parsecs (up to 100 lightyears) – they are the densest and oldest gravitationally bound stellar systems in the Universe. Their central star densities can reach a million times the stellar density near our Sun. About 150 globular clusters orbit the Milky Way but more massive galaxies can have over 10,000 gravitationally bound globular clusters. As their stars have mostly formed at the same time but with different masses, globular clusters are ideal laboratories for studies of stellar dynamics and stellar evolution.

The dynamical evolution of globular clusters, however, is very complex. Unlike in galaxies, the stellar densities are so high that stars can interact in close gravitational encounters or might even physically collide with each other. Because of these interactions there are more tightly bound binary stars than for normal galactic field stars. Moreover, in a process called mass-segregation more massive stars sink to the center of the system.

Search @ KIAA-PKU

Search

Upcoming Events

- *Pulsars and FRBs: Recent Developments*
  Speaker: Richard N. Manchester (CSIRO Astronomy and Space Science, Australia)
  3 Nov 2016 - 4:00pm
  KIAA-PKU Auditorium
- *The role of vortices in multi-dimensional protoplanetary disks*
  Speaker: Hui Li
  7 Nov 2016 - 12:00pm
  DoA, Rm 2907
- *TBD*
  Speaker: Jessy Jose
  8 Nov 2016 - 12:00pm
  KIAA-PKU

more

Navigation

- Biblio

The DRAGON globular cluster simulations: a million stars, black holes and gravitational waves

By shuyan on Mon, 2016-06-27 08:55

An international team of experts from China and Europe has performed the first simulations of globular clusters with a million stars on the high-performance GPU

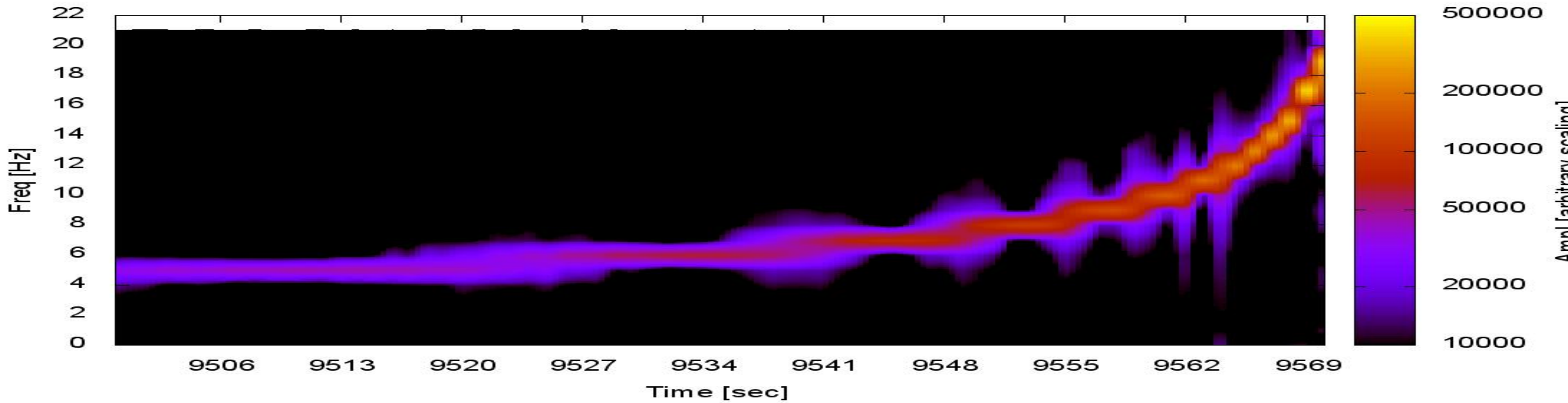
1. ***Wang, Long; Spurzem, Rainer; Aarseth, Sverre; Nitadori, Keigo; Berczik, Peter; Kouwenhoven, M. B. N.; Naab, Thorsten***
   NBODY6++GPU: ready for the gravitational million-body problem
   2015, MNRAS, 450, 4070
   Source
2. ***Wang, Long; Spurzem, Rainer; Aarseth, Sverre; Giersz, Mirek; Askar, Abbas; Berczik, Peter; Naab, Thorsten; M. B. N. Kouwenhoven, Riko Schadow***
   The DRAGON simulations: globular cluster evolution with a million stars

北京大學 PEKING UNIVERSITY

GW Detection Frequency Time Diagram

Top: Our simulation (Wang et al. 2016, Sobolenko et al. In prep.)

Down: Abott et al. 2016 LIGO measurement



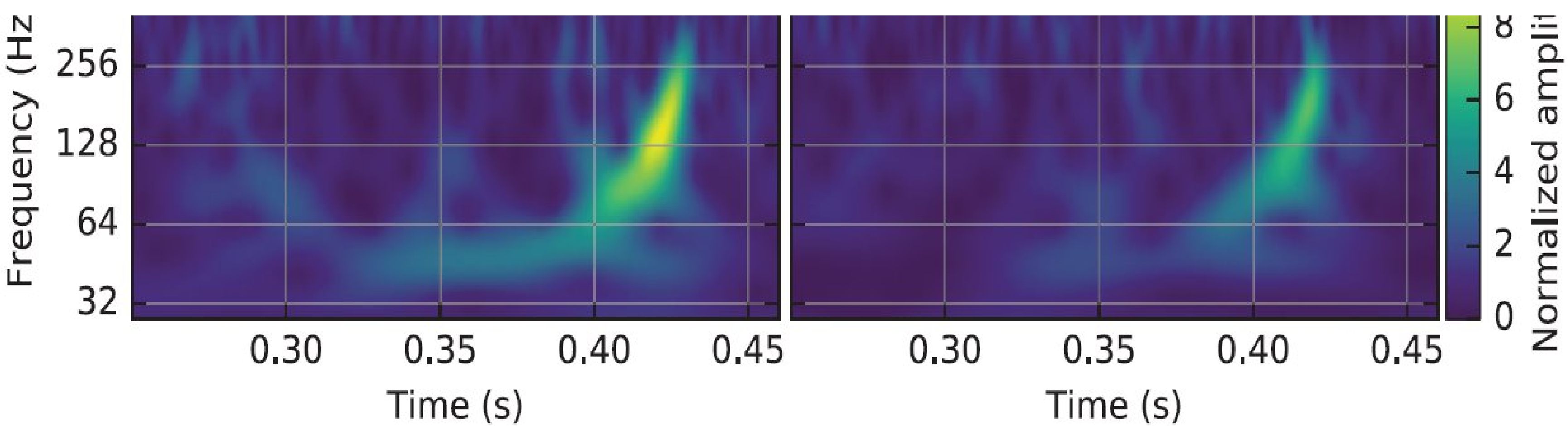


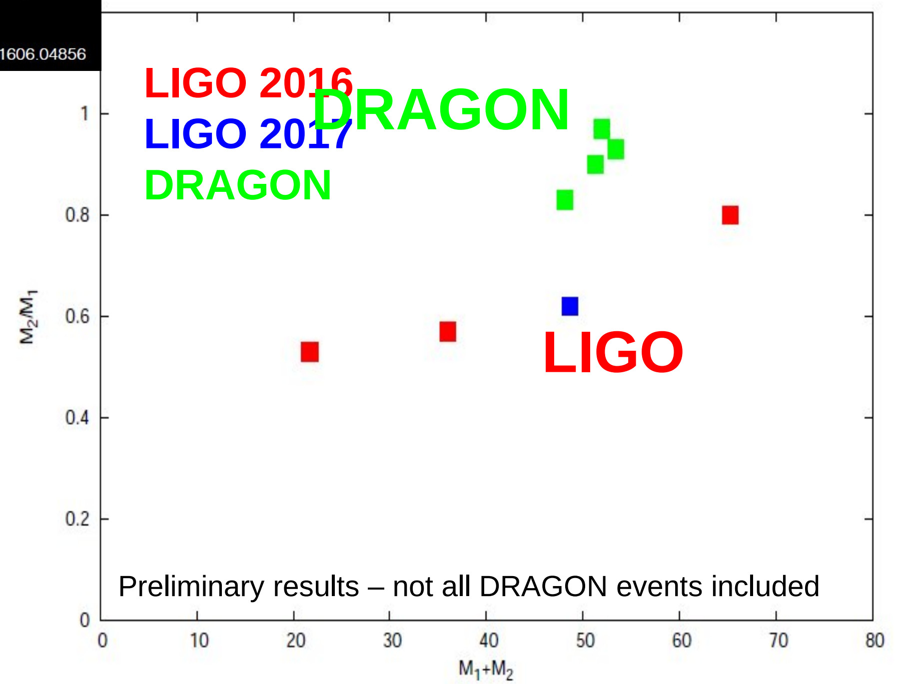
FIG. 1. The gravitational-wave event GW150914 observed by the LIGO Hanford (H1, left column panels) and Livingston (L1, right column panels) detectors. Times are shown relative to September 14, 2015 at 09:50:45 UTC. For visualization, all time series are filtered

The Observed LIGO Events – Table from Brown's Talk at KITP (2)

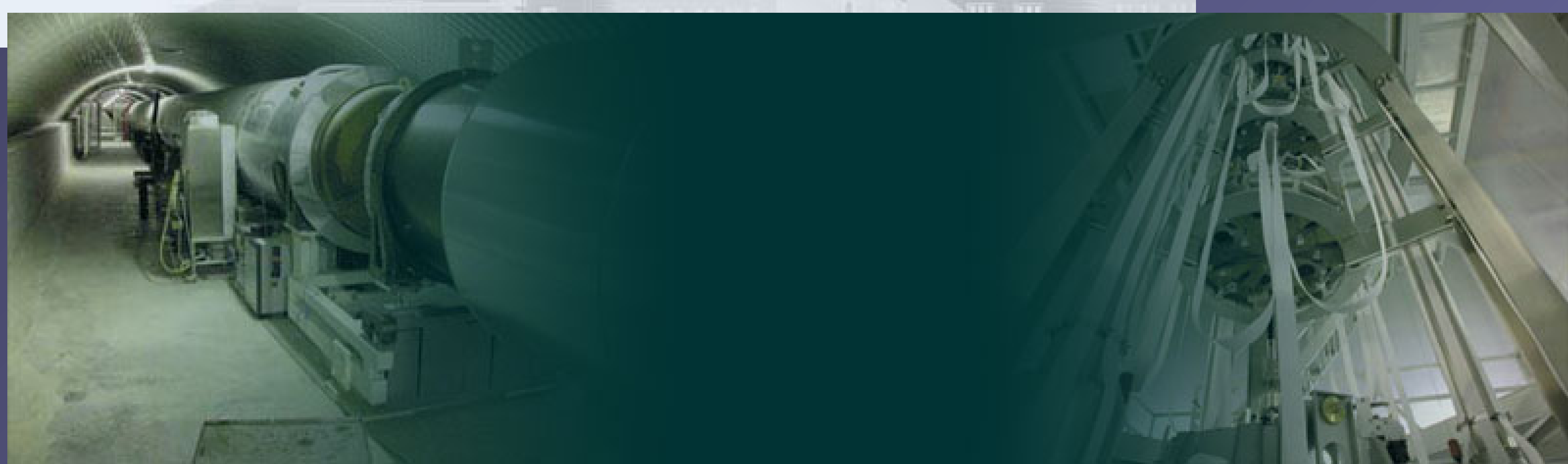
	GW150914	GW151226	LVT151012
Source Mass 1	$36.2^{+5.2}_{-3.8}\ M_\odot$	$14.2^{+8.3}_{-3.7}\ M_\odot$	$23^{+18}_{-6}\ M_\odot$
Source Mass 2	$29.1^{+3.7}_{-4.4}\ M_\odot$	$7.5^{+2.3}_{-2.3}\ M_\odot$	$13^{+4}_{-5}\ M_\odot$
Luminosity Distance	420^{+150}_{-180} Mpc	440^{+180}_{-190} Mpc	1000^{+500}_{-500} Mpc

Abbott, ..., DAB, et al. arXiv:1606.04856

The "Observed" DRAGON Events...



Example: VIRGO Detector in Cascina near Pisa, Italy



VIRGO – Pisa 3km
LIGO – Livingston, LA
Hanford, WA
1km
GEO600 – Hannover
600m
AIGO – Australien
(planned, 5 km)

http://www.ligo-la.caltech.edu/
http://www.ego-gw.it
http://www.geo600.uni-hannover.de

Outreach to 50 Millionen light years (Neutron Stars)

Computational and Computer Science

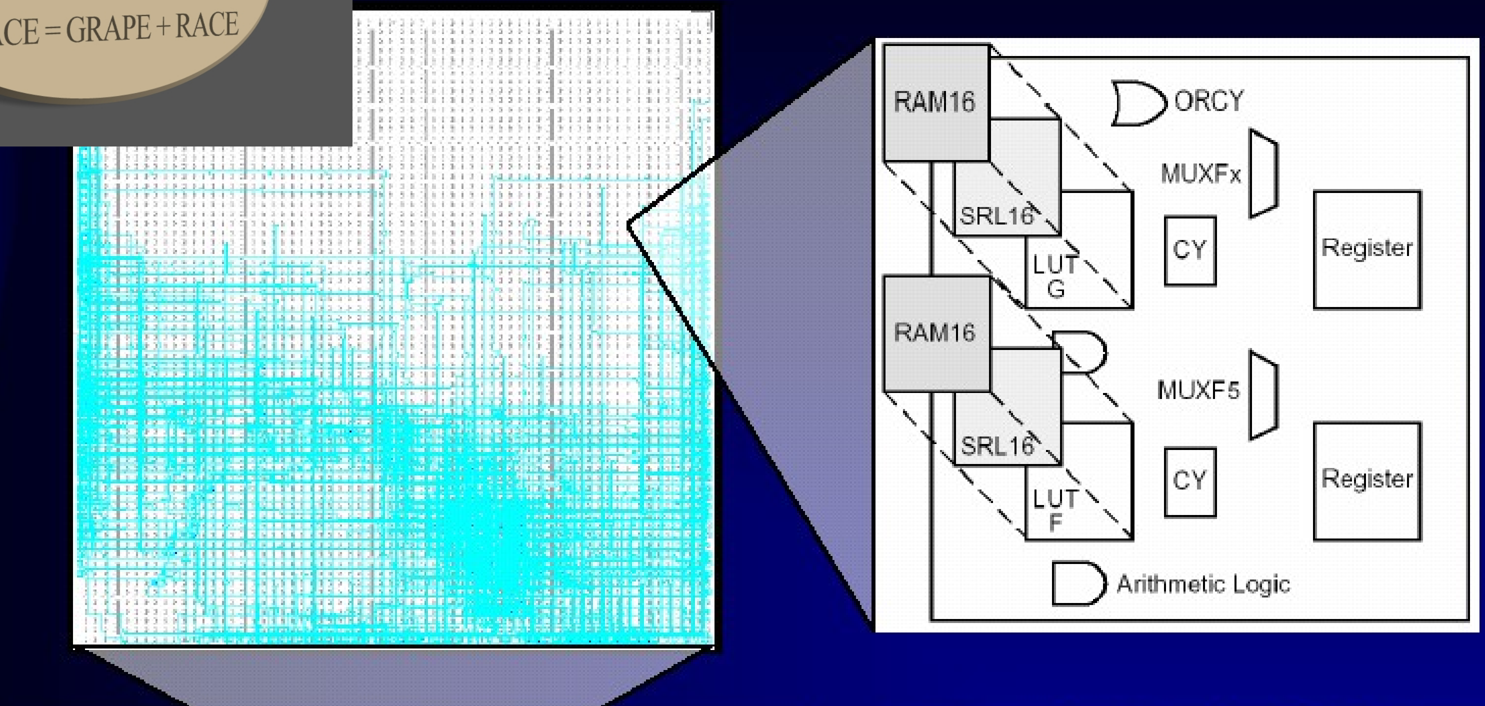
More About the Future

GRACE

GRACE = GRAPE + RACE

FPGA-Plattform MPRACE

2005 - 2009



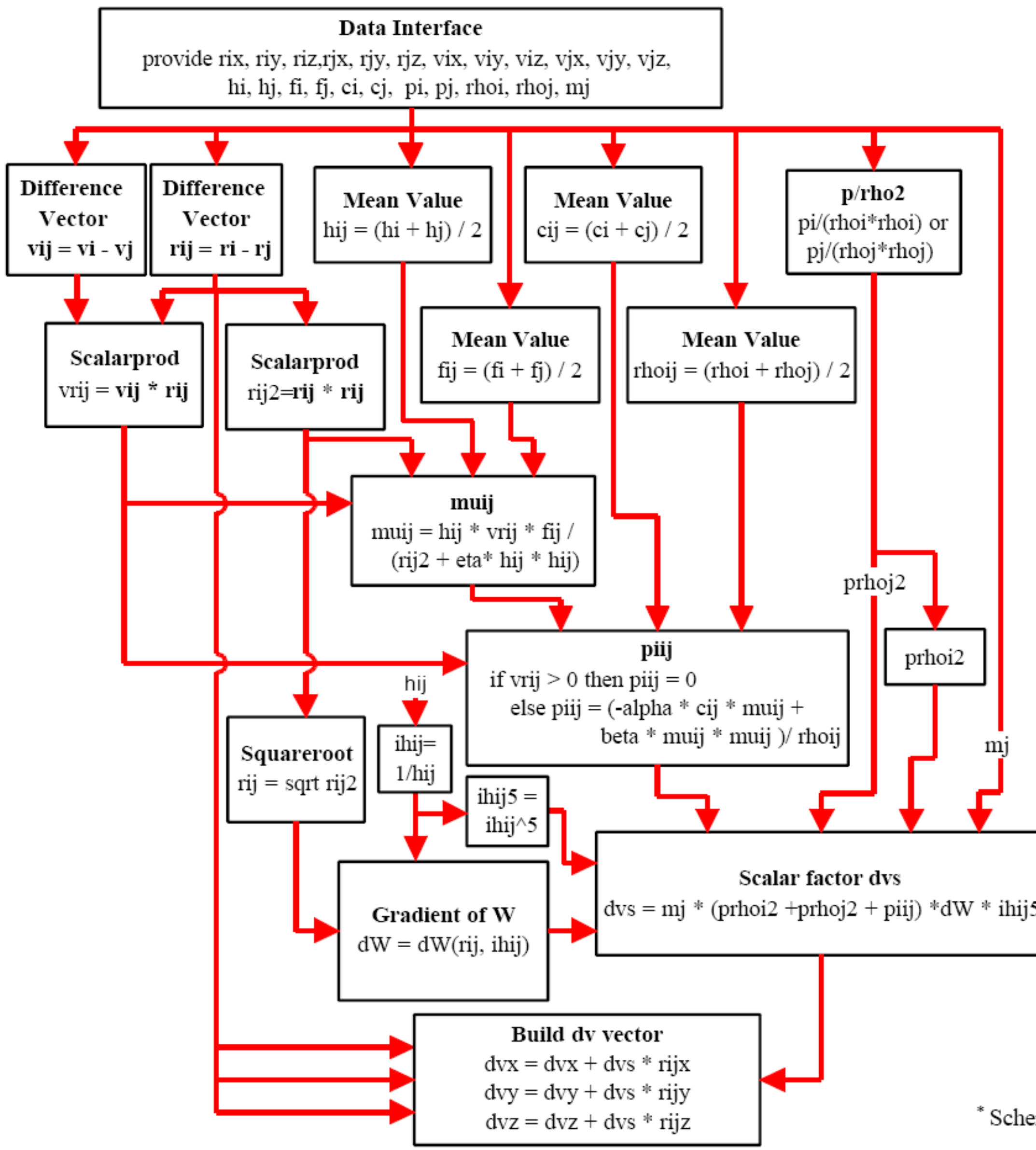
lienhart@ti.uni-mannheim.de

FPGA...

GRACE

GRACE = GRAPE + RACE

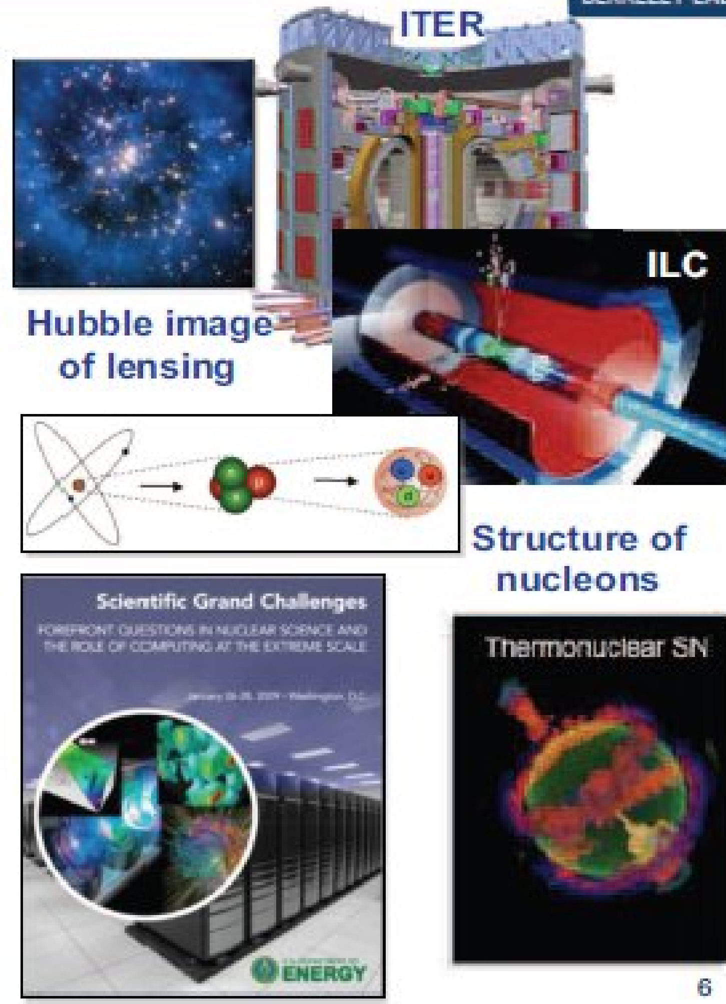
Pressure force pipeline:



* Scheme doesn't show energy term

Exascale simulation will enable fundamental advances in basic science

- High Energy & Nuclear Physics
  - Dark-energy and dark matter
  - Fundamentals of fission fusion reactions
- Facility and experimental design
  - Effective design of accelerators
  - Probes of dark energy and dark matter
  - ITER shot planning and device control
- Materials / Chemistry
  - Predictive multi-scale materials modeling: observation to control
  - Effective, commercial technologies in renewable energy, catalysts, batteries and combustion
- Life Sciences
  - Better biofuels
  - Sequence to structure to function



These breakthrough scientific discoveries and facilities require exascale applications and resources

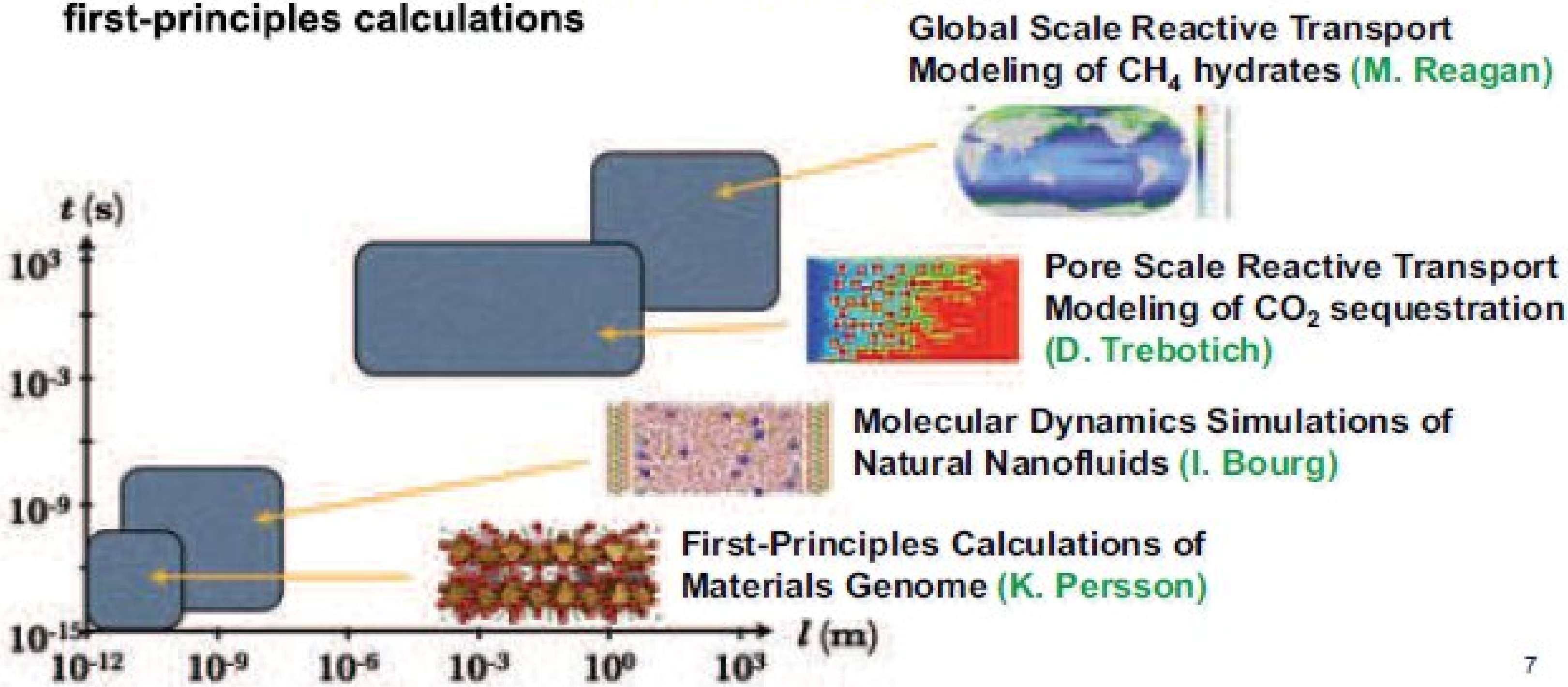
Slide: Horst Simon, Exascale Computing

Advanced Computation in Energy Science at LBNL

Probe natural systems under constraints that are difficult or impossible to impose in the field or laboratory

Reveal the manner in which large-scale phenomena arise from smaller-scale properties

Discover new materials for green technology applications through first-principles calculations

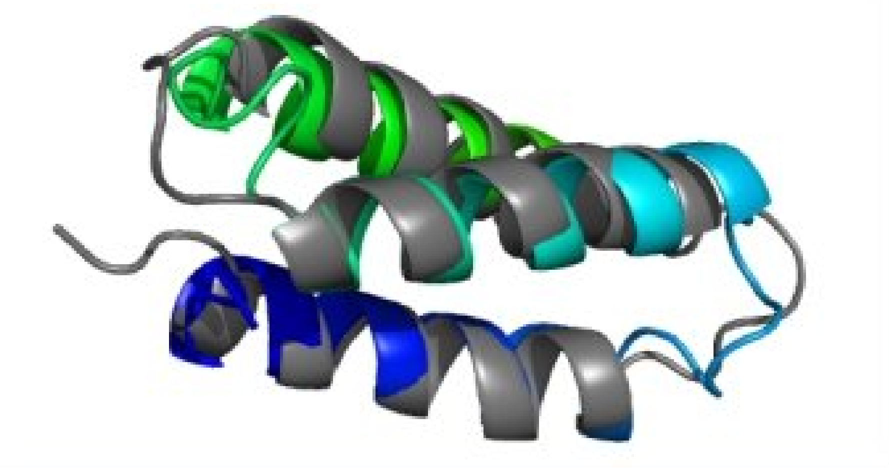


Slide: Horst Simon, Exascale Computing

Research

http://www.fz-juelich.de/ias/jsc/EN/Research/research_node.html

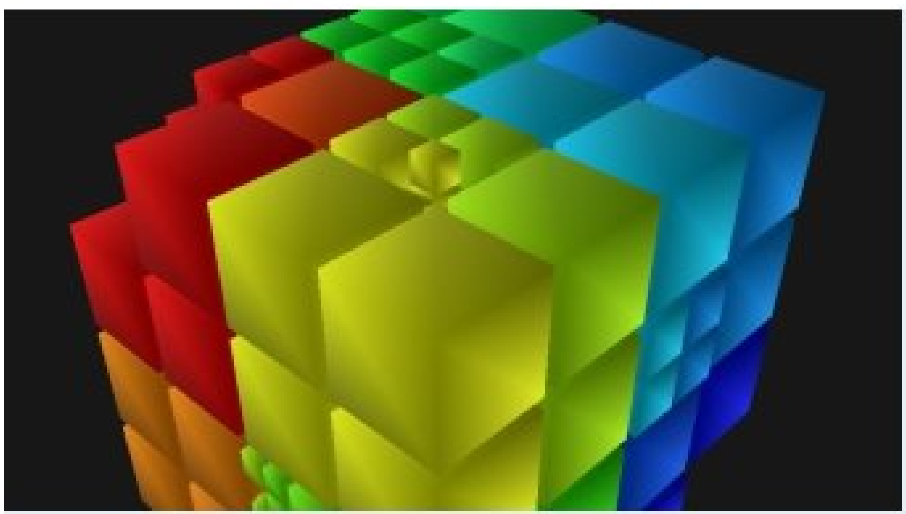
JSC's research and development concentrates on mathematical modelling and numerical, especially parallel algorithms for quantum chemistry, molecular dynamics and Monte-Carlo simulations. The focus in the computer sciences is on cluster computing, performance analysis of parallel programs, visualization, computational steering and grid computing.



Modelling and Simulation

The simulation of complex systems in natural science or engineering depends on the development of adequate mathematical models. Thus the development of realistic and yet efficient models is a core activity at JSC. Examples of simulations are:

- Computational Plasma Physics
- Protein Folding
- Quantum Information Processing
- Civil Security and Traffic



Algorithms and Methods

Efficient simulations need powerful algorithms and methods. JSC focusses on the development of the following methods:

- Fast Coulomb Solvers
- Parallel-In-Time Integration
- Fast Multipole Method
- Parallel I/O

HPCI: High Performance Computing Infrastructure

- Established as Japanese integrated high performance computing infrastructure in 2011
- Variety of computer systems are connected via high speed academic backbone network and provided as ***HPCI*** resources to users in ***Japan and overseas,*** Also it will be a platform for international collaborations.

FY2017 Allocated computing resources

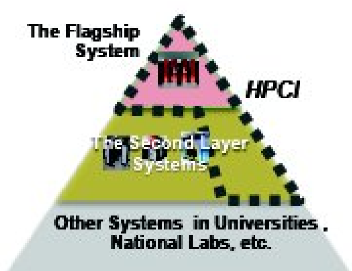
~9.6 PFlops* x *Yr.

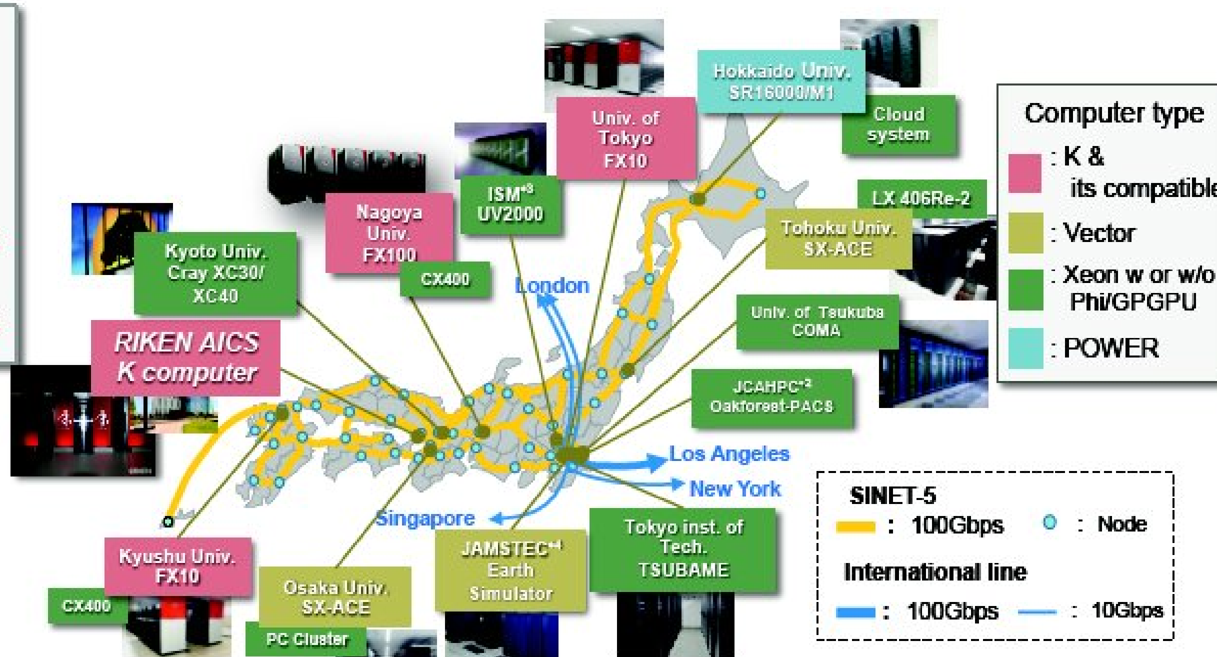
K computer

~4 PFlops* x *Yr.

Others in total

~5.6 PFlops* x *Yr.





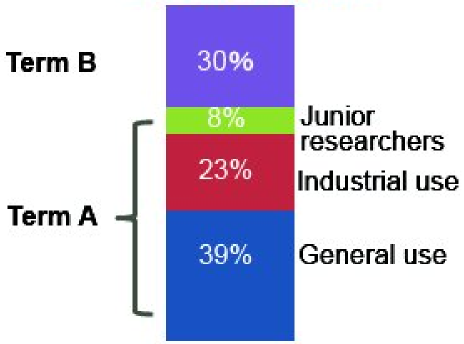
*2 : Joint Center for Advanced High Performance Computing
*3 : The Institute of Statistical Mathematics
*4 : Japan Agency for Marine-Earth Science and Technology

Resources allocation and Awarding results of FY 2017

RIST

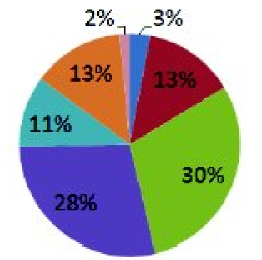
■ *K computer*

~ 4 PFlops・year (corresponding to 45**%** of total K resource)



Submitted	96
Awarded	67
Ratio	70%

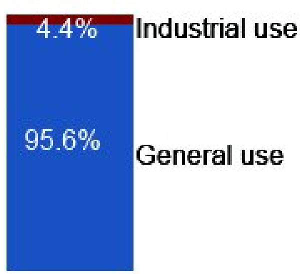
Major Research areas



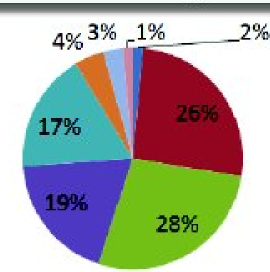
- Physics and space physics
- Material science and chemistry
- Engineering and manufacturing
- Bio and life science
- Environment, disaster prevention and mitigation

■ Other *HPCI* system

~ 5.6 PFlops・year



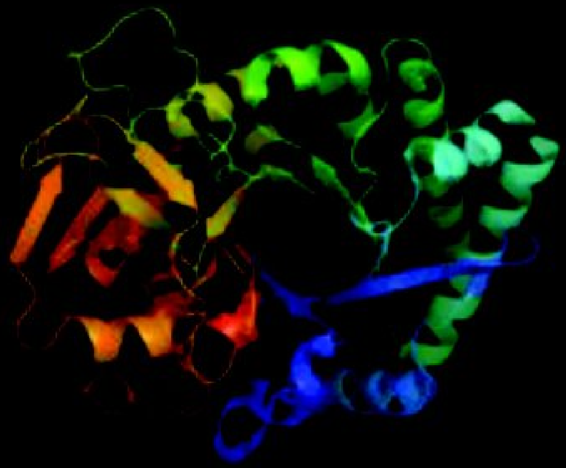
Submitted	155
Awarded	69
Ratio	45%



Deep Learning Is Getting Real Now …

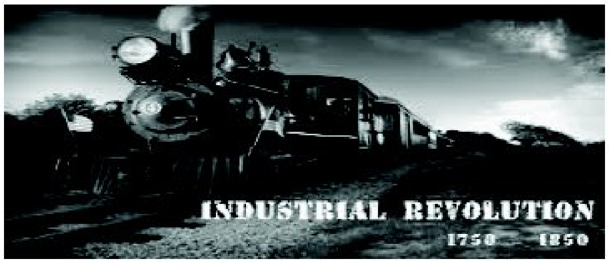
Deep learning algorithm does as well as dermatologists in identifying skin cancer

Artificial intelligence could build new drugs faster than any human team

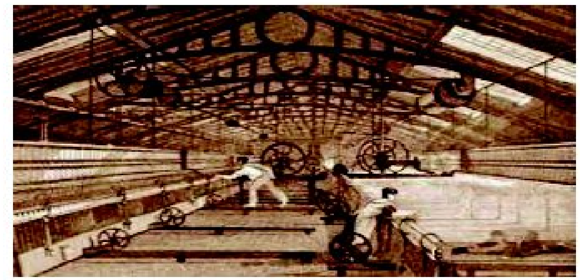


AMD | RADEON

MILS: Machine Intelligence Led Services



Mills

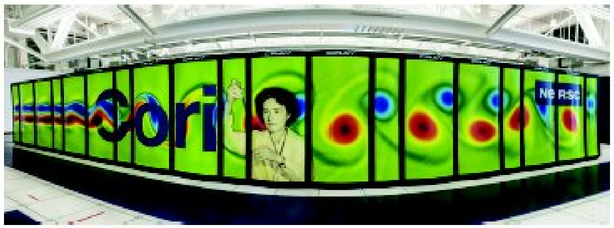


MILS

Intelligence Too Big for a Single Machine

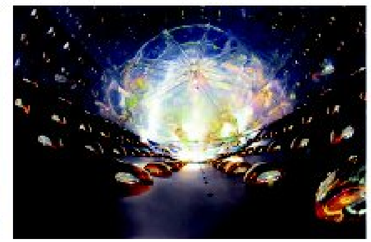
"We're seeing a rebirth of artificial intelligence driven by the cloud, huge amounts of data and the learning algorithms of software,"
Larry Smarr, founding director of the California Institute for Telecommunications and Information Technology
http://bits.blogs.nytimes.com/2014/06/11/intelligence-too-big-for-a-single-machine/

Deep Learning in Science

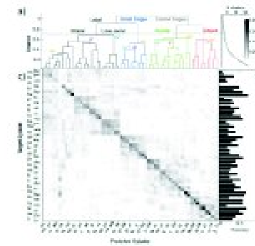


Cray XC40 system at NERSC

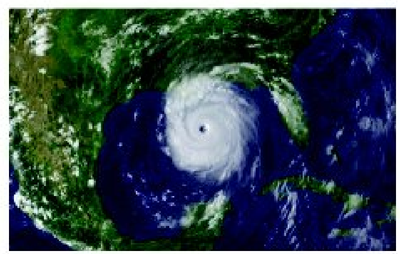
Modeling galaxy shapes



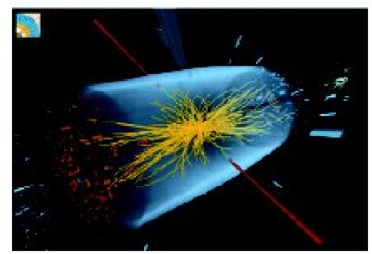
Clustering Daya Bay events



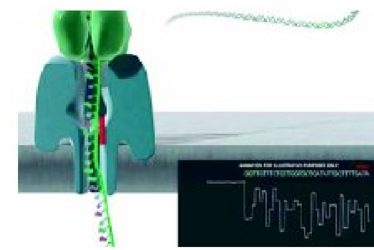
Decoding speech from ECoG



Detecting extreme weather



Classifying LHC events



Oxford Nanopore sequencing

NERSC

BERKELEY LAB
LAWRENCE BERKELEY NATIONAL LABORATORY

Opportunities to apply DL widely in support of classic HPC simulation and modelling

...the new hype: HPC and AI … here by T. Maruyama, ISC17:

Processor Designed for Deep Learning

FUJITSU

K computer

Utilizing technologies derived from the K computer

FY2018 ～

DLU™

(Deep Learning Unit)

Features of DLU

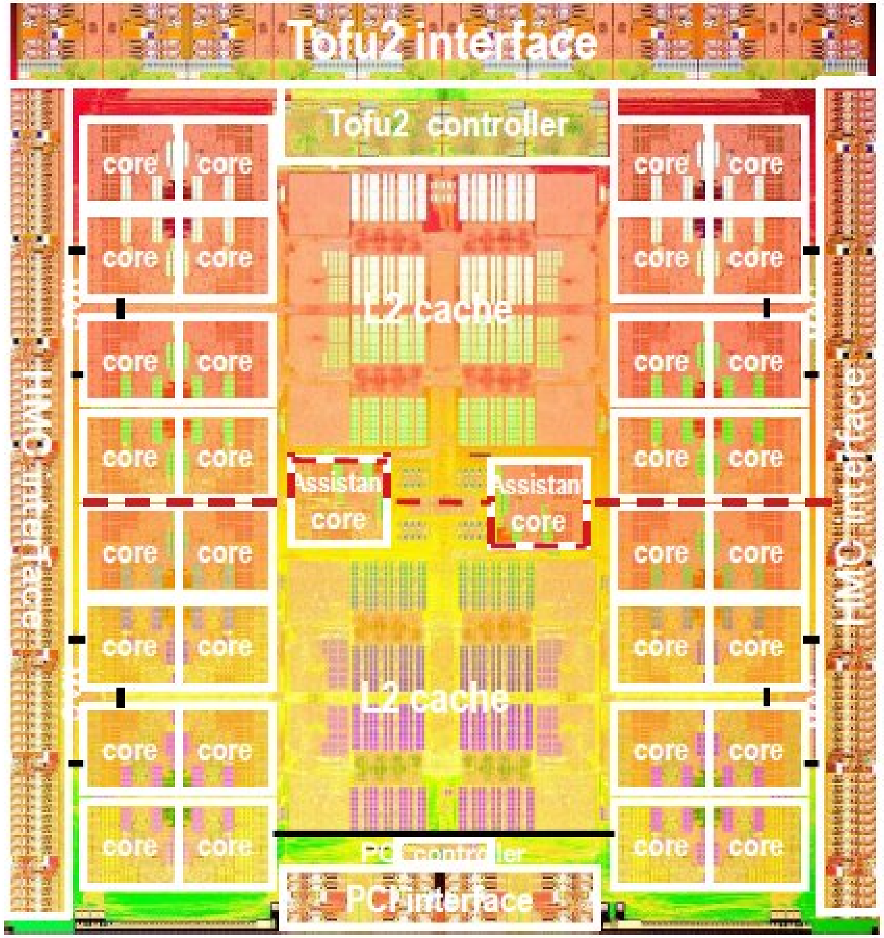
- Architecture designed for Deep Learning
- Low power consumption design
- Optimized precision

→ Goal: 10x Performance / Watt compared to competitors

- Scalable design with Tofu interconnect technology

→ Ability to handle large-scale neural networks

SPARC64™ XIfx Chip (HPC)

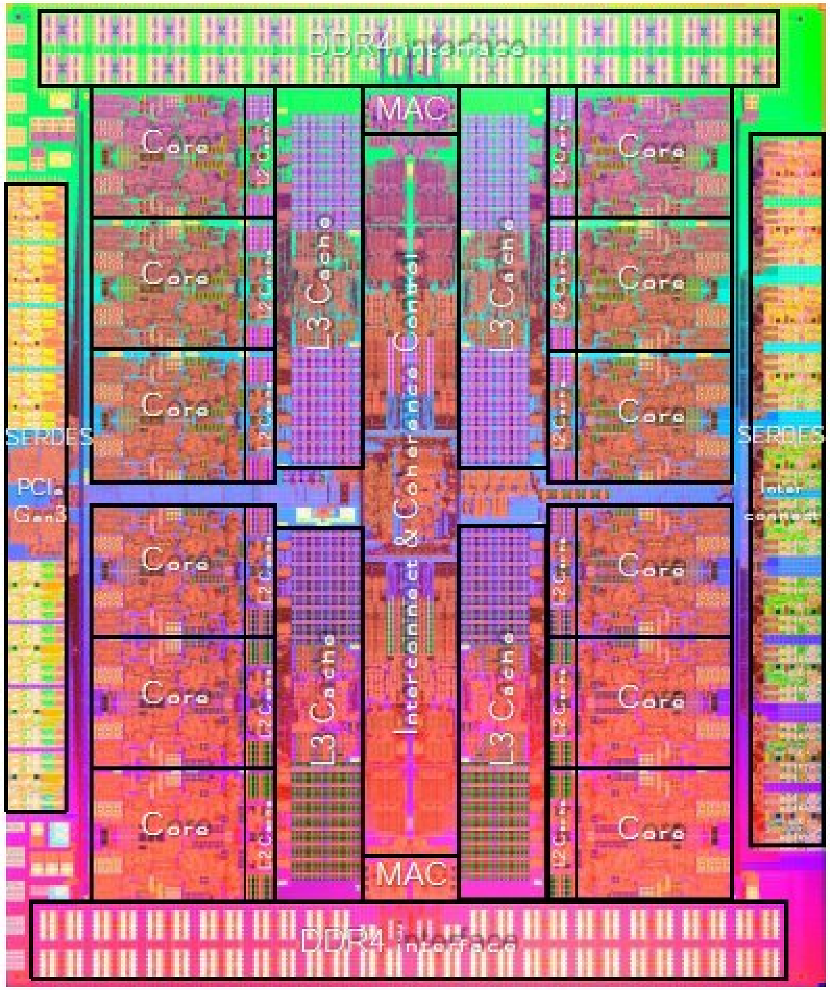


Many (32+2) cores, Medium CPU GHz

- Architecture Features
  - 32 computing cores + 2 assistant cores
  - HPC-ACE2 (256bit SIMD) Fujitsu's ISA enhancements
  - Sector Cache: Cache with SW controllability
  - 24 MB L2 cache
- 20nm CMOS
  - 3,750M transistors
  - 2.2GHz
- Performance (peak)
  - 1.1TFlops
  - HMC 240GB/s x 2 (in/out)
  - Tofu2 125GB/s x 2 (in/out)

SPARC64™ XII Chip (UNIX)

FUJITSU

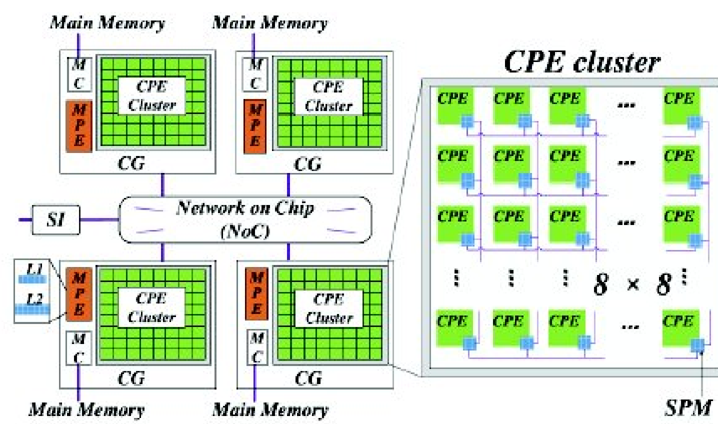


Multiple big cores, High CPU GHz

- Architecture Features
  - 12 cores x 8 threads
  - SWoC ("Software on Chip")
    Fujitsu's ISA enhancements
  - 32MB L3 cache
  - Embedded MAC and IOC
- 20nm CMOS
  - 25.8mm x 30.8mm
  - 5,450M transistors
  - 4.25GHz (up to 4.35GHz with "High Speed Mode" enabled)
- Performance (peak)
  - 417GIPS / 835GFlops
  - 153GB/s memory throughput

SUNWAY TAIHULIGHT

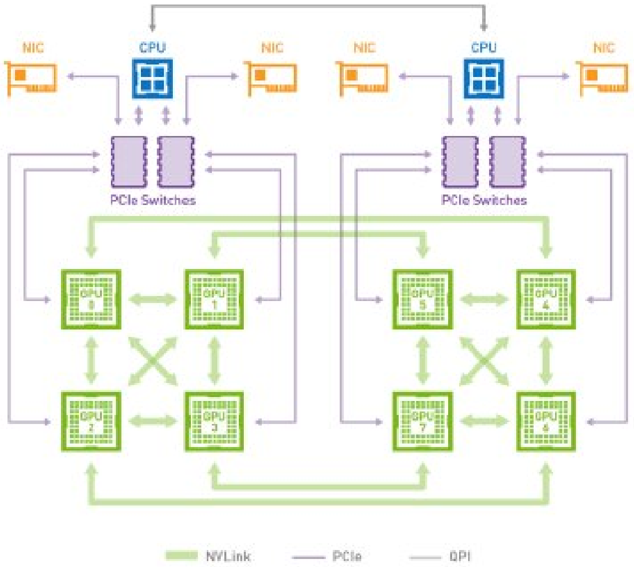
- SW26010 processor (Chinese design, ISA, & fab)
- 1.45 GHz
- Node = 260 Cores (1 socket)
  - 4 – core groups
  - 32 GB memory
- 40,960 nodes in the system
- 10,649,600 cores total
- 1.31 PB of primary memory (DDR3).
- 125.4 Pflop/s theoretical peak
- 93 Pflop/s HPL, 74% peak
- 15.3 Mwatts water cooled
- 3 of the 6 finalists for Gordon Bell Award@SC16



SYSTEMS APPROACHES TO EXASCALE

More GPUs, Fewer CPUs:

- Titan: 1GPU/CPU
- Summit: 3 GPUs/CPU
- Exascale: ?



Faster Serial Processing (~~MANY CORE~~):

- Run 8x Fewer Cores @ 2x Speed

Denser Packaging:

- Move Networking to Faster Local Networks: NVLINK

EXASCALE: "50X FASTER THAN TITAN"

Per-GPU -hardware- speedups will be less than 50x

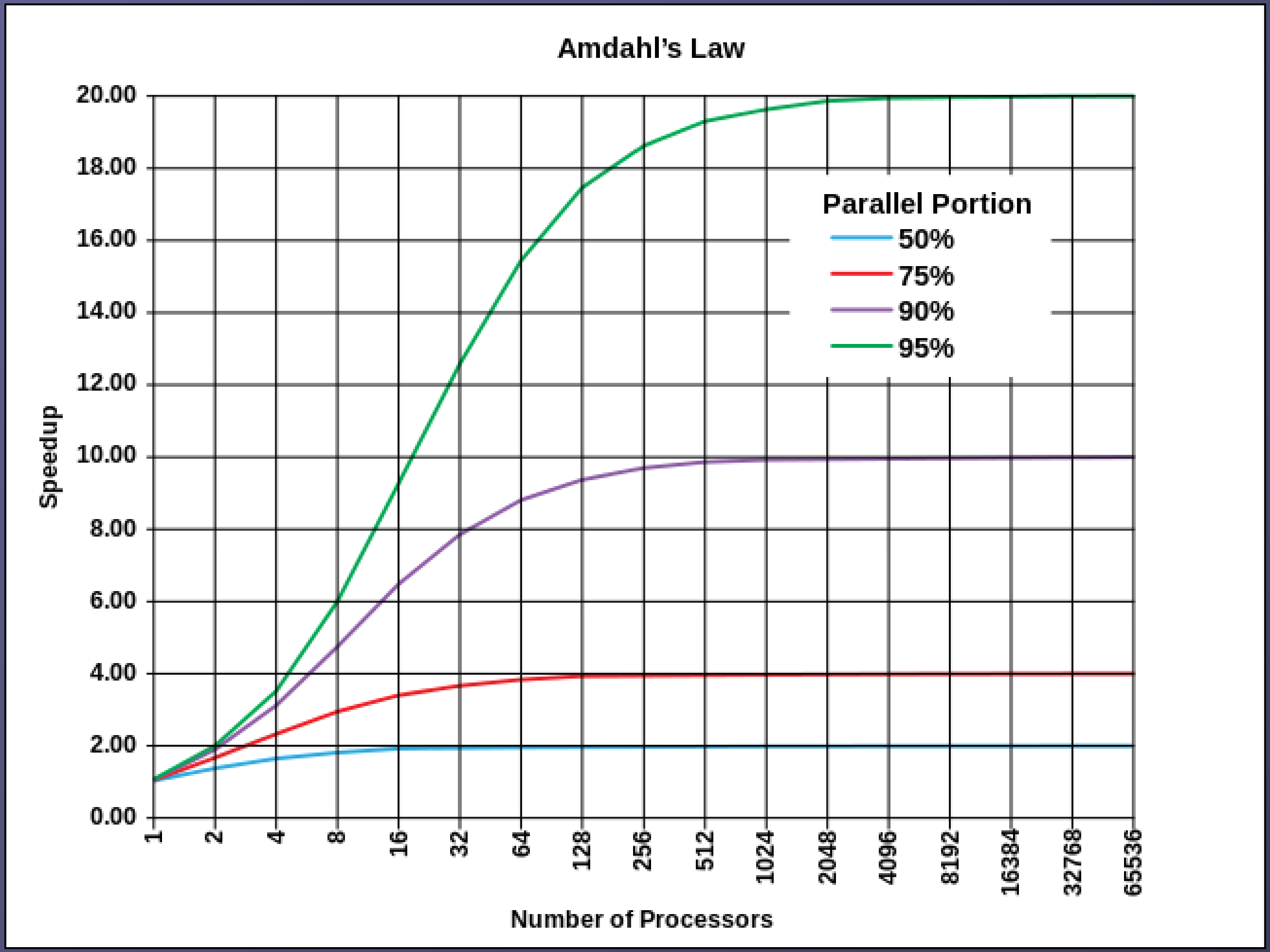
	2013 Kepler	2016 Pascal	2017 Volta	2021*	Speedup
FP64 Tflop/s	1.5	4.5	7	7-21	5-15
Memory GB/s	288	720	900	900-4000	3-14
I/O BW GB/s	7	80	150	150-500	20-70
Deep Learning FP16 Tflop/s	3	20	112	112-500	37-166
Deep Learning BW GB/s	576	2880	3600	3600-16000	6-27

*Extremely Fuzzy Public Projections for 2021

Parallel Computing

Some basic ideas

Amdahl's Law (Gene Amdahl 1967)



Evolution according to Amdahl's law of the theoretical speedup of the execution of a program in function of the number of processors executing it, for different values of p. The speedup is limited by the serial part of the program. For example, if 95% of the program can be parallelized, the theoretical maximum speedup using parallel computing would be 20 times.

Calculate Amdahl's Law:

Let X be the part of my program (in terms of computing time) which can be parallelised. The sequential computing time Tseq is normalized to unity (1), and can be expressed as:

Tseq = 1 = X + (1-X)

The parallel computing time Tpar under ideal conditions (ideal load balancing, ultrafast communication):

Tpar = X/p + (1-X) with processor number (core number) p

Then the speed-up of the program S = Tseq / Tpar :

S = 1 / (1-X+X/p) ; Note: Tpar/Tseq = 1/S (sometimes also plotted)

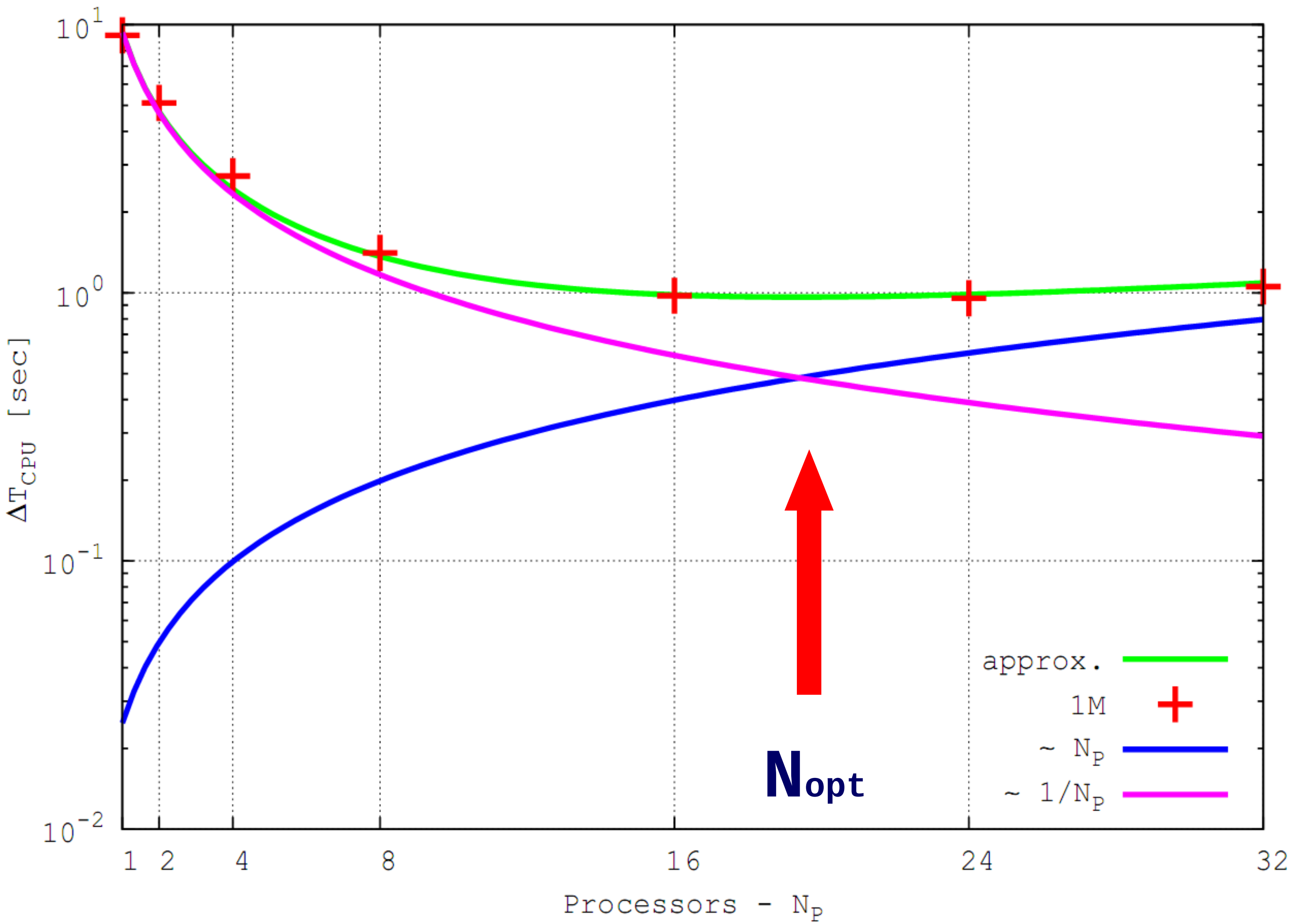
Note the limit if p is very large: S = 1/(1-X). And if X ~ 1: S ~ p

With communication overhead:

Tpar = X/p + (1-X) + Tcomm → **S = 1 / (1-X+X/p+Tcomm)**

If Tcomm independent of p we have for large p: S = 1 / (1-X + Tcomm) = const.

Parallel code on cluster



Strong and Soft Scaling

- ➔ Strong Scaling: Fixed Problem size, increase p
- ➔ Soft Scaling: Increase Problem size, increase p (constant amount of work per processing element)

Ansatz for Soft Scaling:

- ➔ **Tseq = p = p (X + (1-X))**
- ➔ **Tpar = X + p (1-X)**
- ➔ **S = Tseq/Tpar = p / (X+p (1-X))**

If X~1: S = p ; Tpar = X = const.

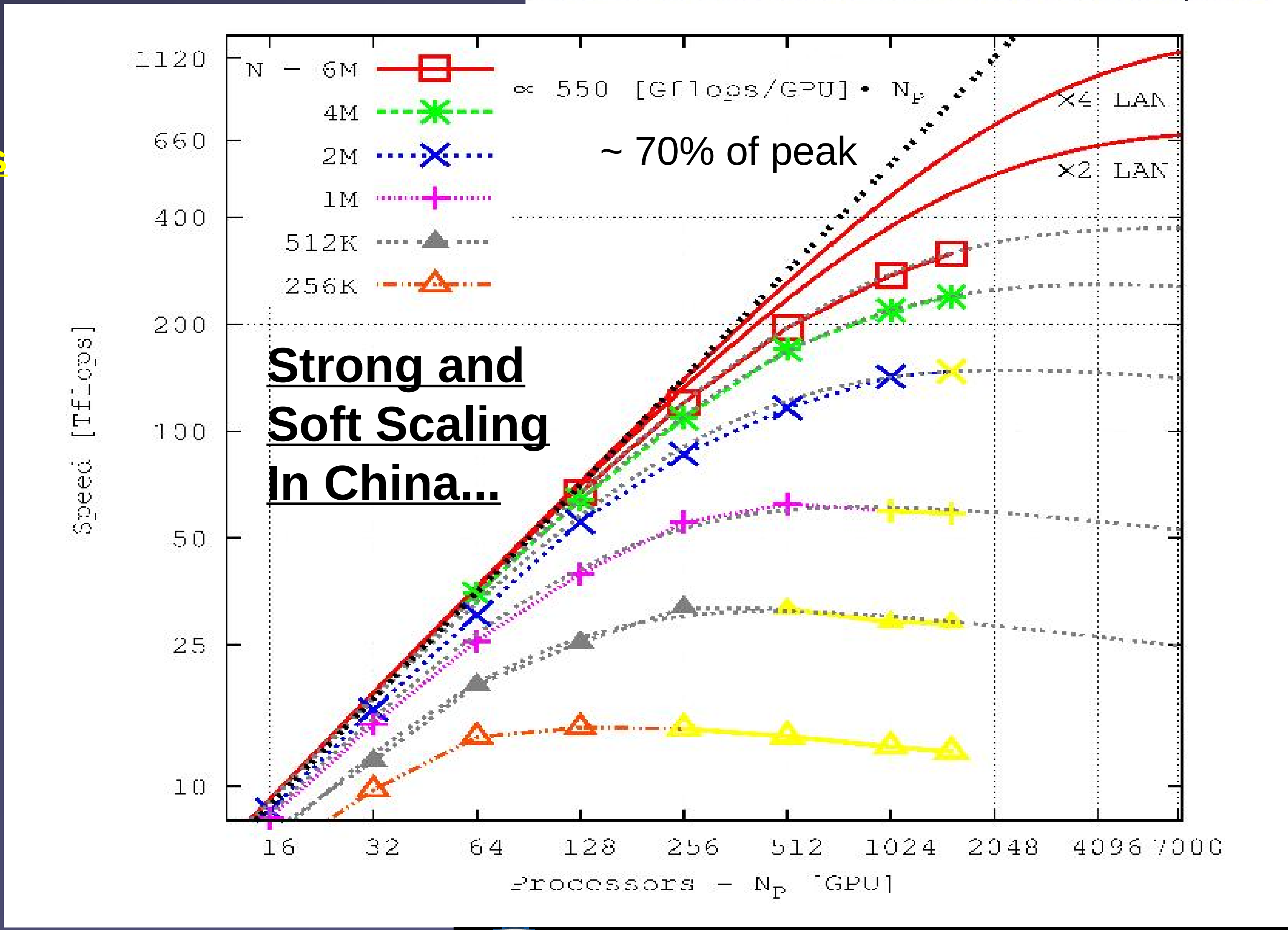
ΦGPU – NBODY Code

中国科学院国家天文台
National Astronomical Observatories, CAS

350 Teraflop/s
1600 GPUs .
440 cores
= 704.000
GPU-Cores

Using
Mole-8.5
of
IPE/CAS
Beijing

Berczik et al.
2013



中国科学院过程工程研究所
Institute Of Process Engineering, Chinese Academy Of Sciences

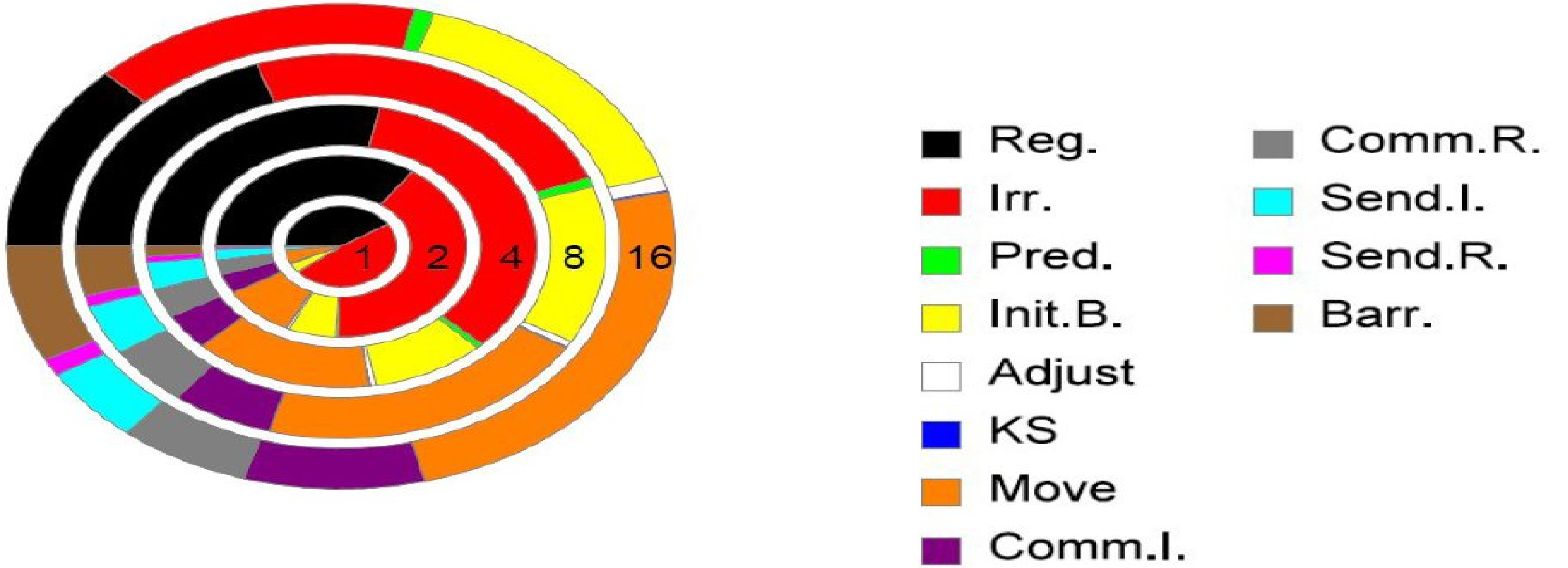
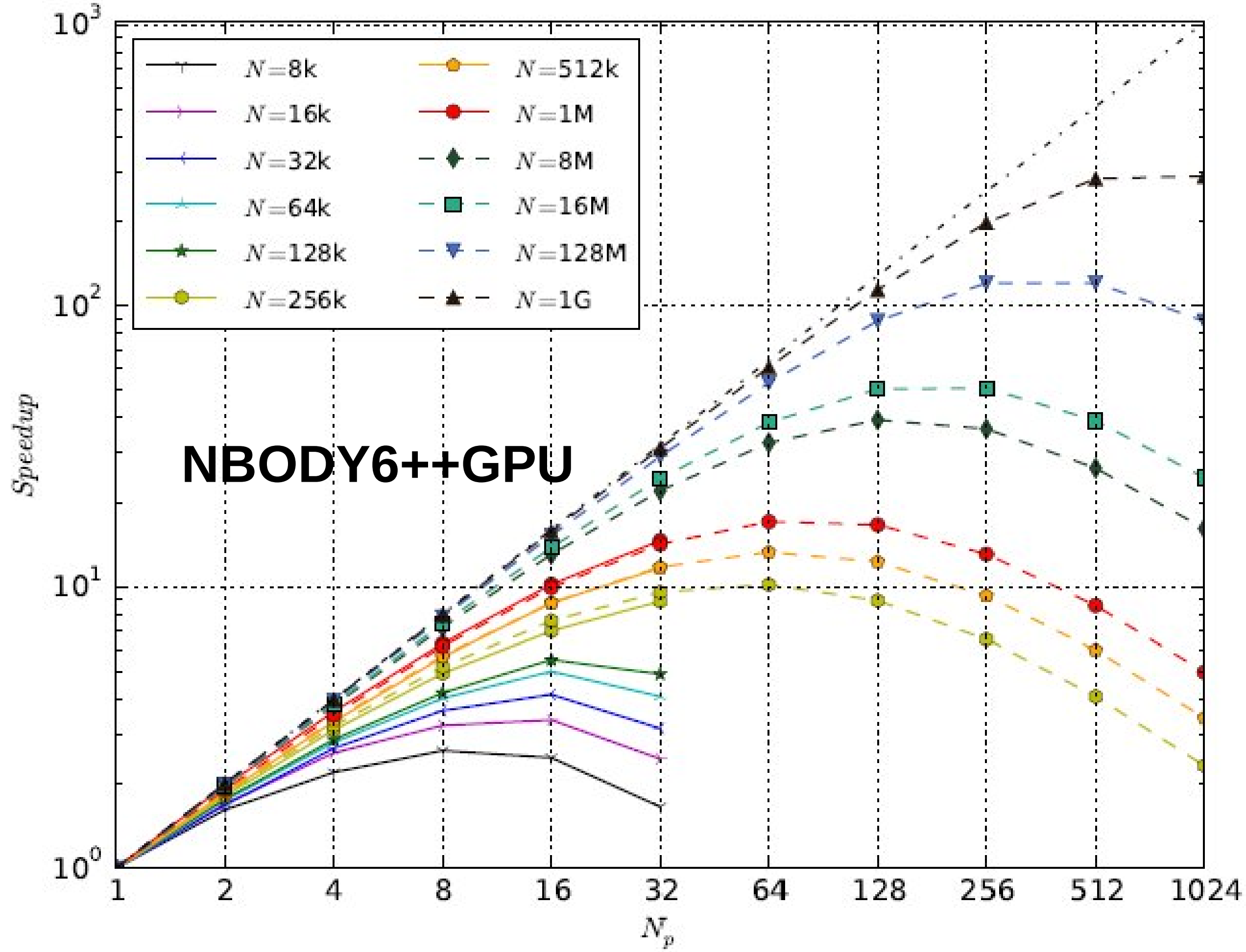


Table 1 Main components of NBODY6++

Description	Timing variable	Expected scaling N	Expected scaling N_p	Fitting value [sec]
Regular force computation	$T_{\rm reg}$	$\mathcal{O}(N_{\rm reg} \cdot N)$	$\mathcal{O}(N_p^{-1})$	$(2.2 \cdot 10^{-9} \cdot N^{2.11} + 10.43) \cdot N_p^{-1}$
Irregular force computation	$T_{\rm irr}$	$\mathcal{O}(N_{\rm irr} \cdot \langle N_{nb} \rangle)$	$\mathcal{O}(N_p^{-1})$	$(3.9 \cdot 10^{-7} \cdot N^{1.76} - 16.47) \cdot N_p^{-1}$
Prediction	$T_{\rm pre}$	$\mathcal{O}(N^{kn_p})$	$\mathcal{O}(N_p^{-kp_p})$	$(1.2 \cdot 10^{-6} \cdot N^{1.51} - 3.58) \cdot N_p^{-0.5}$
Data moving	$T_{\rm mov}$	$\mathcal{O}(N^{kn_{m1}})$	$\mathcal{O}(1)$	$2.5 \cdot 10^{-6} \cdot N^{1.29} - 0.28$
MPI communication (regular)	$T_{\rm mcr}$	$\mathcal{O}(N^{kn_{cr}})$	$\mathcal{O}(kp_{cr} \cdot \frac{N_p-1}{N_p})$	$(3.3 \cdot 10^{-6} \cdot N^{1.18} + 0.12)(1.5 \cdot \frac{N_p-1}{N_p})$
MPI communication (irregular)	$T_{\rm mci}$	$\mathcal{O}(N^{kn_{ci}})$	$\mathcal{O}(kp_{ci} \cdot \frac{N_p-1}{N_p})$	$(3.6 \cdot 10^{-7} \cdot N^{1.40} + 0.56)(1.5 \cdot \frac{N_p-1}{N_p})$
Synchronization	$T_{\rm syn}$	$\mathcal{O}(N^{kn_s})$	$\mathcal{O}(N_p^{kp_s})$	$(4.1 \cdot 10^{-8} \cdot N^{1.34} + 0.07) \cdot N_p$
Sequential parts on host	$T_{\rm host}$	$\mathcal{O}(N^{kn_h})$	$\mathcal{O}(1)$	$4.4 \cdot 10^{-7} \cdot N^{1.49} + 1.23$



Huang, Berczik, Spurzem, Res. Astron. Astroph. 2016, 16, 11.

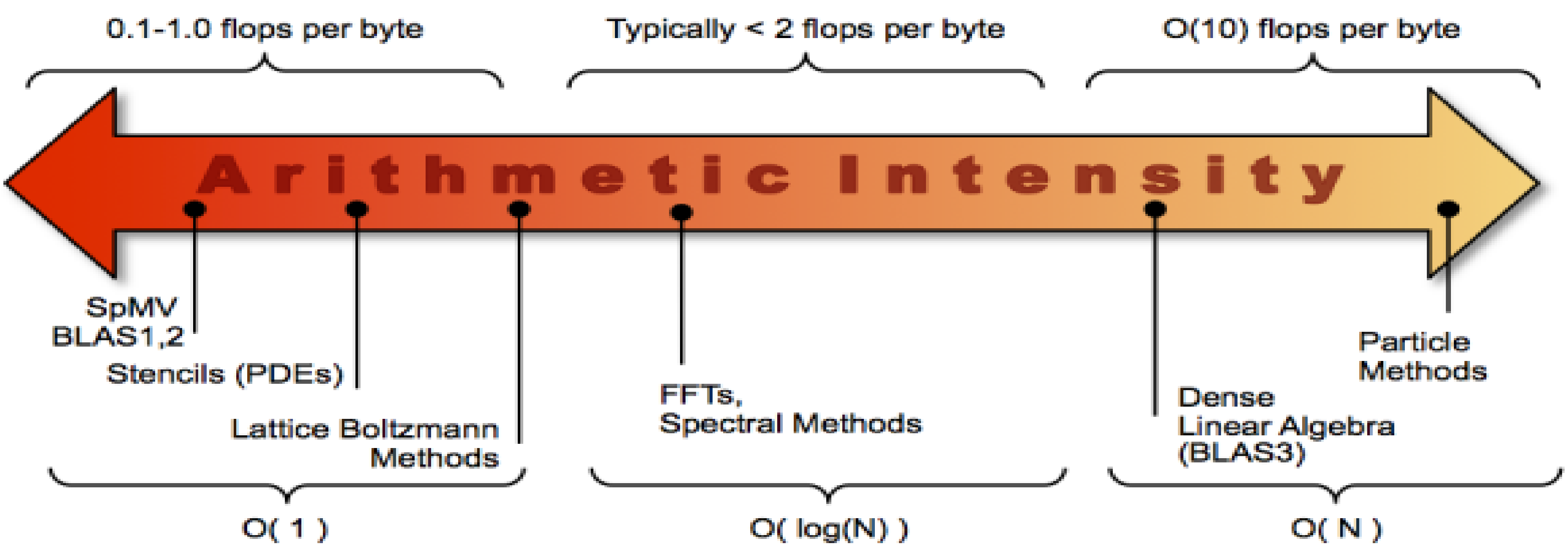
Fig. 2 The speed-up (S) of NBODY6++ as a function of particle number (N) and processor number (N_p). Solid points are the measured speed-up ratio between sequential and parallel wall-clock time, dash lines predict the performance of larger scale simulations further. The symbols used in figure have the magnitudes: $1k = 1,024$, $1M = 1k^2$ and $1G = 1k^3$.

Roofline Performance Model (LBL)

http://crd.lbl.gov/departments/computer-science/PAR/research/roofline

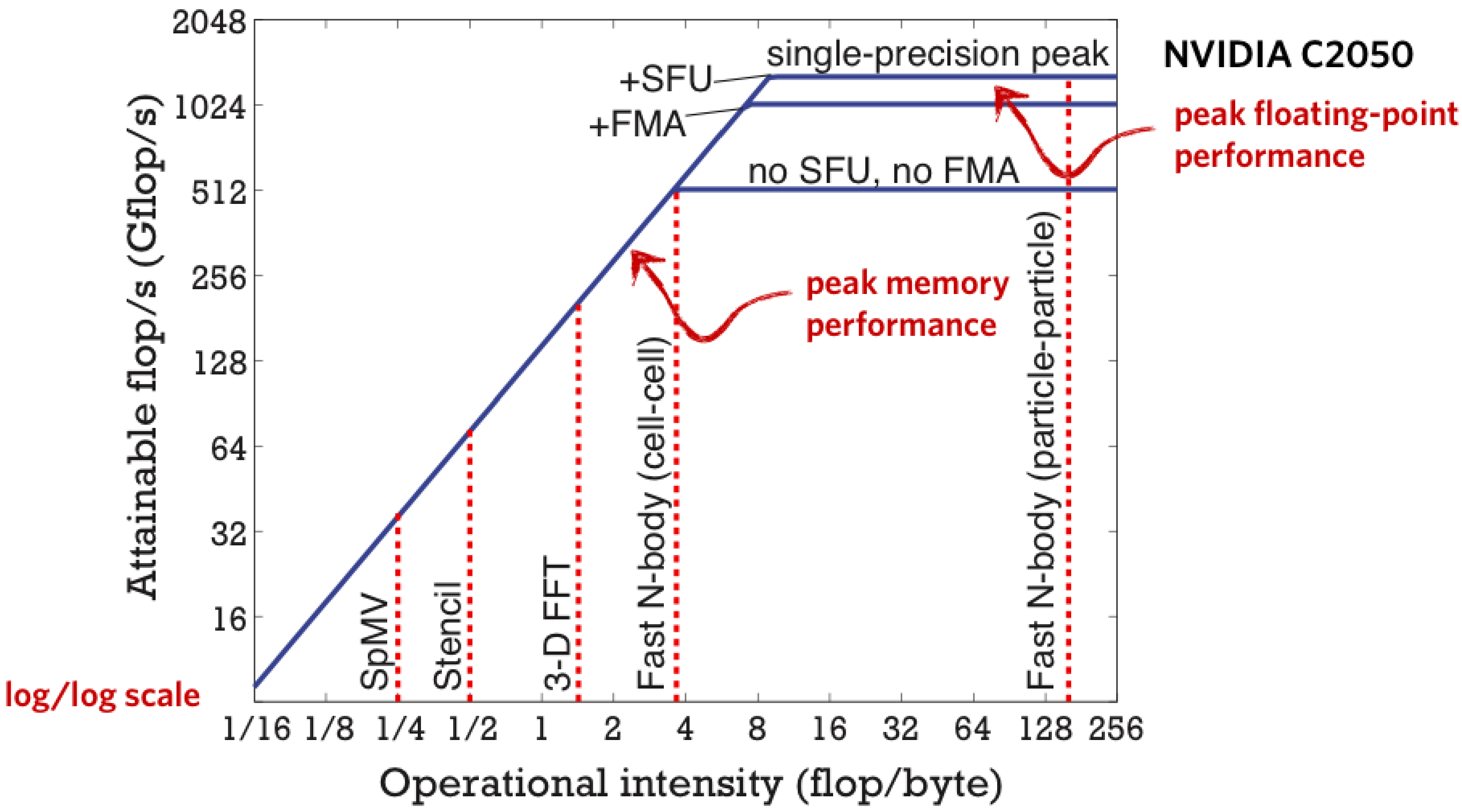
Arithmetic Intensity

The core parameter behind the Roofline model is Arithmetic Intensity. Arithmetic Intensity is the ratio of total floating-point operations to total data movement (bytes).



Roofline Performance Model (LBL)

http://lorenabarba.com/wp-content/uploads/2012/01/roofline_slide.png



Parallel Computing

Matrix Multiply and Debugging

Timing with CUDA Event API

```
int main ()
{
    cudaEvent_t start, stop;
    float time;

    cudaEventCreate (&start);
    cudaEventCreate (&stop);

    cudaEventRecord (start, 0);

    someKernel <<<grids, blocks, 0, 0>>> (...);

    cudaEventRecord (stop, 0);
    cudaEventSynchronize (stop);

    cudaEventElapsedTime (&time, start, stop);

    cudaEventDestroy (start);
    cudaEventDestroy (stop);

    printf ("Elapsed time %f sec\n", time*.001);

    return 1;
}
```

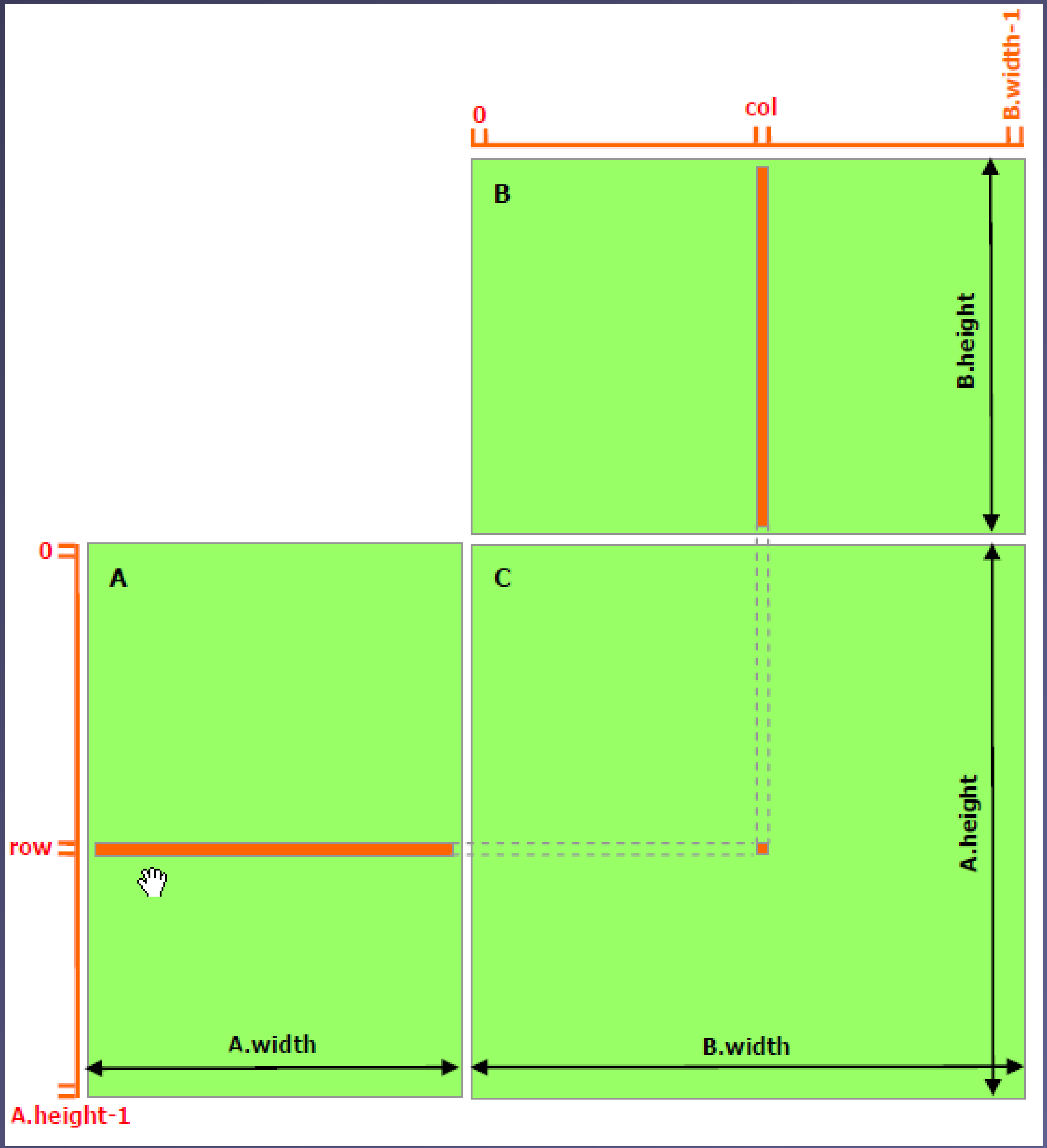
CUDA Event API Timer are,

- OS independent
- High resolution
- Useful for timing asynchronous calls

← Ensures kernel execution has completed (cudaEventSynchronize)

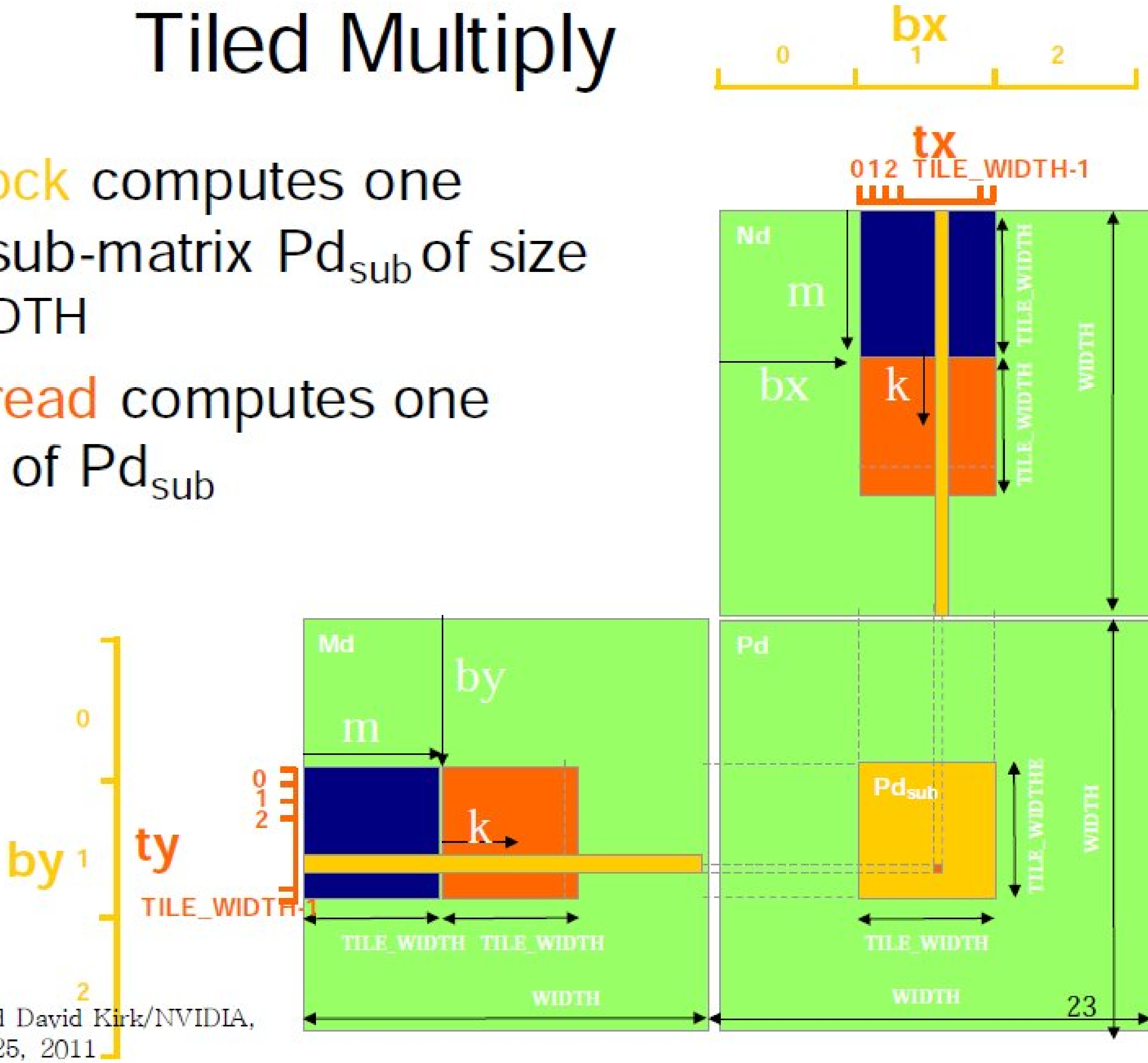
Standard CPU timers will not measure the timing information of the device.

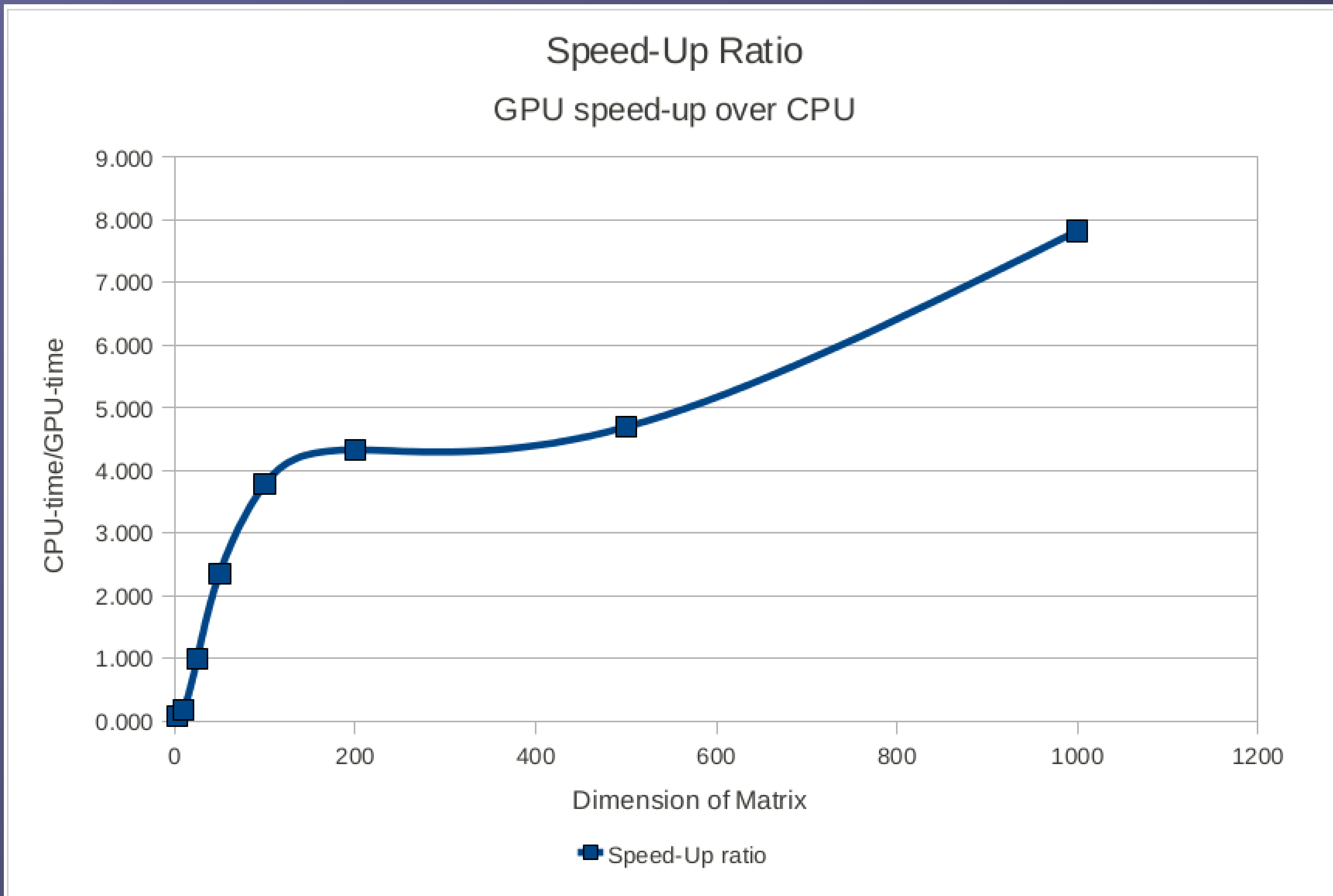
Intuitive multiply



Tiled Multiply

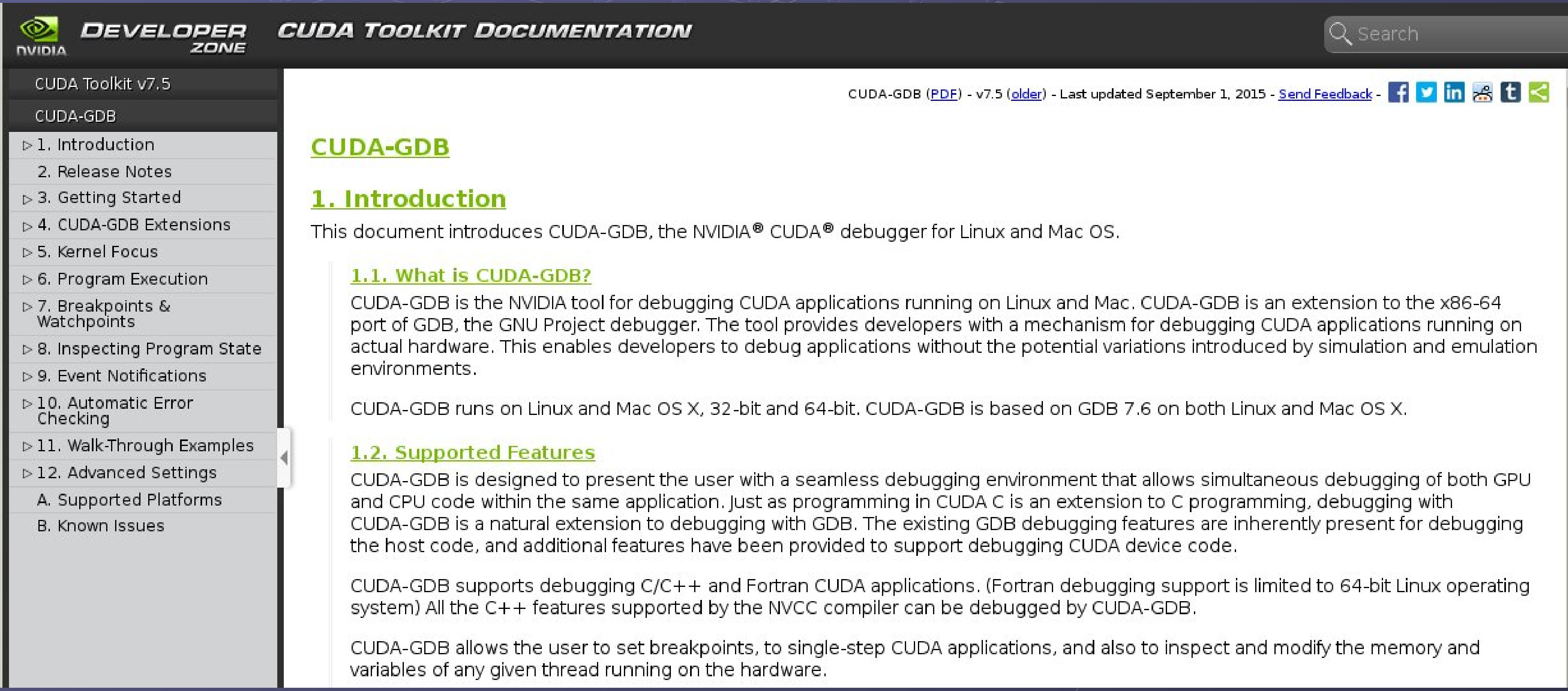
- Each block computes one square sub-matrix Pd_{sub} of size TILE_WIDTH
- Each thread computes one element of Pd_{sub}




Speed-Up Ratio
GPU speed-up over CPU
9.000
8.000
7.000
6.000
5.000
4.000
3.000
2.000
1.000
0.000
CPU-time/GPU-time
0
200
400
600
800
1000
1200
Dimension of Matrix
Speed-Up ratio

CUDA – GNU Debugger – CUDA-gdb

http://docs.nvidia.com/cuda/cuda-gdb/index.html



NVIDIA DEVELOPER ZONE CUDA TOOLKIT DOCUMENTATION Search

CUDA Toolkit v7.5

CUDA-GDB

- ▷ 1. Introduction
- 2. Release Notes
- ▷ 3. Getting Started
- ▷ 4. CUDA-GDB Extensions
- ▷ 5. Kernel Focus
- ▷ 6. Program Execution
- ▷ 7. Breakpoints & Watchpoints
- ▷ 8. Inspecting Program State
- ▷ 9. Event Notifications
- ▷ 10. Automatic Error Checking
- ▷ 11. Walk-Through Examples
- ▷ 12. Advanced Settings
- A. Supported Platforms
- B. Known Issues

CUDA-GDB (PDF) - v7.5 (older) - Last updated September 1, 2015 - Send Feedback -

CUDA-GDB

1. Introduction

This document introduces CUDA-GDB, the NVIDIA® CUDA® debugger for Linux and Mac OS.

1.1. What is CUDA-GDB?

CUDA-GDB is the NVIDIA tool for debugging CUDA applications running on Linux and Mac. CUDA-GDB is an extension to the x86-64 port of GDB, the GNU Project debugger. The tool provides developers with a mechanism for debugging CUDA applications running on actual hardware. This enables developers to debug applications without the potential variations introduced by simulation and emulation environments.

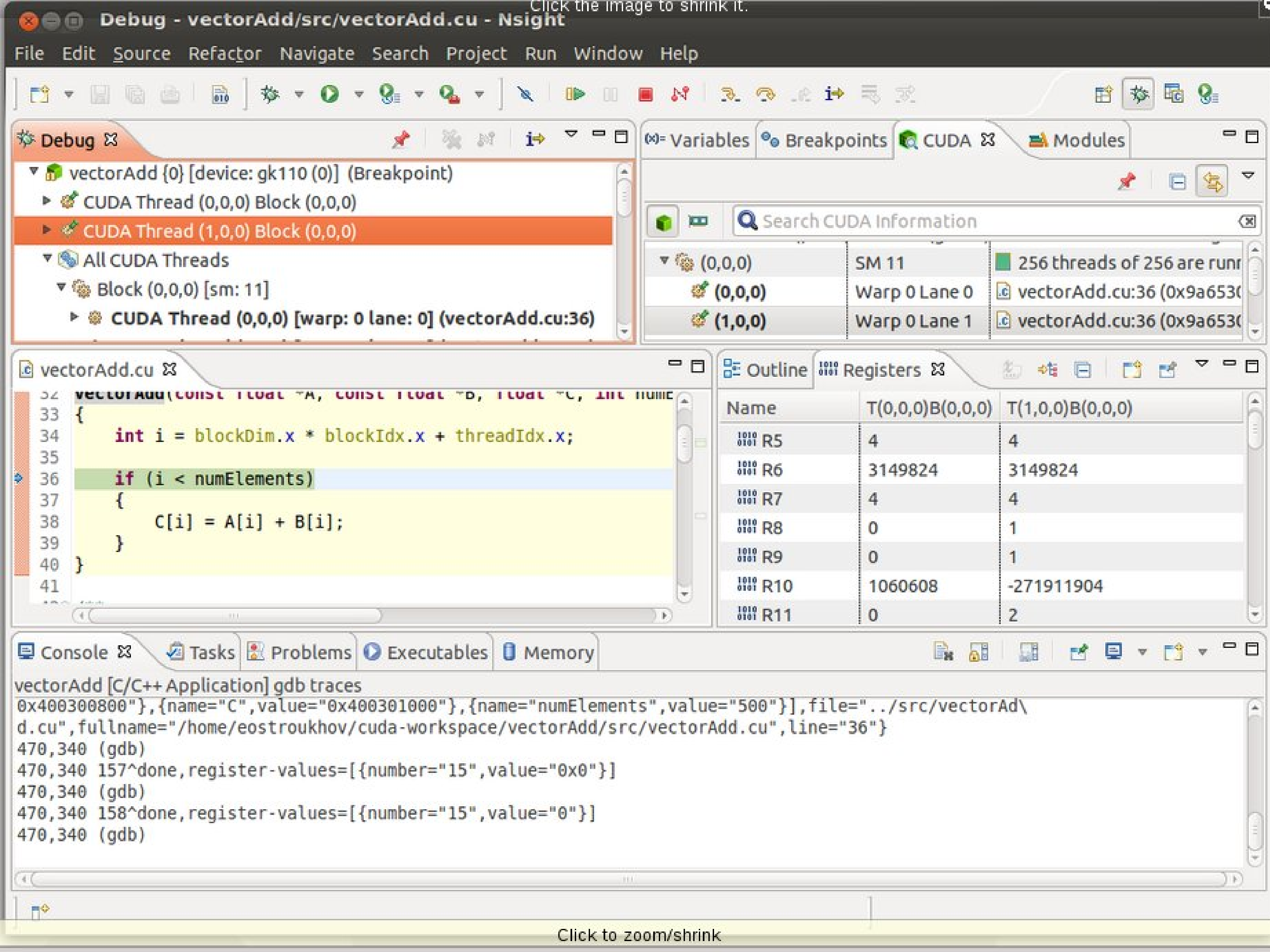
CUDA-GDB runs on Linux and Mac OS X, 32-bit and 64-bit. CUDA-GDB is based on GDB 7.6 on both Linux and Mac OS X.

1.2. Supported Features

CUDA-GDB is designed to present the user with a seamless debugging environment that allows simultaneous debugging of both GPU and CPU code within the same application. Just as programming in CUDA C is an extension to C programming, debugging with CUDA-GDB is a natural extension to debugging with GDB. The existing GDB debugging features are inherently present for debugging the host code, and additional features have been provided to support debugging CUDA device code.

CUDA-GDB supports debugging C/C++ and Fortran CUDA applications. (Fortran debugging support is limited to 64-bit Linux operating system) All the C++ features supported by the NVCC compiler can be debugged by CUDA-GDB.

CUDA-GDB allows the user to set breakpoints, to single-step CUDA applications, and also to inspect and modify the memory and variables of any given thread running on the hardware.


Click the image to shrink it.
Debug - vectorAdd/src/vectorAdd.cu - Nsight
File Edit Source Refactor Navigate Search Project Run Window Help
Debug
vectorAdd {0} [device: gk110 (0)] (Breakpoint)
CUDA Thread (0,0,0) Block (0,0,0)
CUDA Thread (1,0,0) Block (0,0,0)
All CUDA Threads
Block (0,0,0) [sm: 11]
CUDA Thread (0,0,0) [warp: 0 lane: 0] (vectorAdd.cu:36)
Variables
Breakpoints
CUDA
Modules
Search CUDA Information
(0,0,0) SM 11 256 threads of 256 are runn
(0,0,0) Warp 0 Lane 0 vectorAdd.cu:36 (0x9a653
(1,0,0) Warp 0 Lane 1 vectorAdd.cu:36 (0x9a653
vectorAdd.cu
{
int i = blockDim.x * blockIdx.x + threadIdx.x;
if (i < numElements)
{
C[i] = A[i] + B[i];
}
}
Outline
Registers
Name T(0,0,0)B(0,0,0) T(1,0,0)B(0,0,0)
R5 4 4
R6 3149824 3149824
R7 4 4
R8 0 1
R9 0 1
R10 1060608 -271911904
R11 0 2
Console
Tasks
Problems
Executables
Memory
vectorAdd [C/C++ Application] gdb traces
0x400300800"},{name="C",value="0x400301000"},{name="numElements",value="500"}],file="../src/vectorAd\
d.cu",fullname="/home/eostroukhov/cuda-workspace/vectorAdd/src/vectorAdd.cu",line="36"}
470,340 (gdb)
470,340 157^done,register-values=[{number="15",value="0x0"}]
470,340 (gdb)
470,340 158^done,register-values=[{number="15",value="0"}]
470,340 (gdb)
Click to zoom/shrink

Wrapping Up 1

Exercises (CUDA Lectures in afternoon)

1. hello, device- first kernel call, hello world, GPU properties
2. add - vector addition using one thread in one block only
3. add-index - vector addition using blocks in parallel, one thread per block only.
4. add-parallel - vector addition using all blocks and threads in parallel
5. dot - scalar product using shared memory of one block only for reduction
6. dot-full - scalar product using shared memory and atomic add across blocks
7. histo - histogram using fat threads and atomic add on shared and global memory, timing
8. dot-perfect - scalar product using fat threads, shared memory, final reduction on host.
9. matmul - matrix multiplication with tiled access shared memory

Wrapping Up 2

Elements of CUDA C learnt:

threadId.x , blockId.x, blockDim.x, gridDim.x	Threads, Blocks
(threadId.y, blockId.y, blockdim.y, gridDim.y	work with 2D grids)
kernel<<<n,m>>> (...)	kernel calls
__device__ __global__	device code
__shared__	shared memory on GPU
cudaMalloc / cudaFree	manage global memory of GPU
cudaMemcpy / cudaMemset	copy/set to or from memory
cudaGetDeviceProperties	get device properties in program
cudaEventCreate, cudaEventRecord, cudaEventSynchronize, cudaEventElapsedTime, cudaEventDestroy	CUDA profiling
AtomicAdd	atomic functions

Wrapping Up 3

What we have not yet learnt...

__constant__	constant memory on GPU
cudaBindTexture	using texture memory
fat threads for 2D and 3D stencils	thread coalescence opt.
cudaStreamCreate, cudaStreamDestroy	working with CUDA streams

Additional deeper material:

Lectures by Prof. Wen-Mei Hwu Chicago in Berkeley 2012 and Beijing 2013, see **http://iccs.lbl.gov/workshops/tutorials.html**

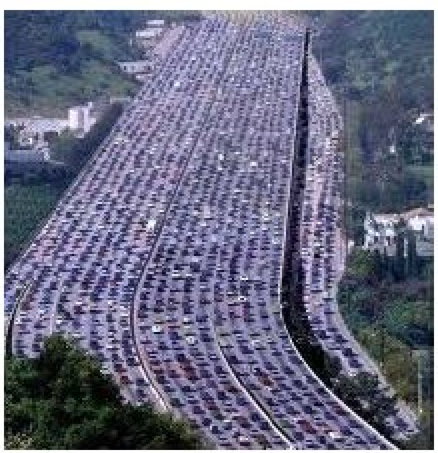
(down on page links to all lecture files, also available on request from spurzem@nao.cas.cn)

Lecture1: Computational thinking
Lecture2: Parallelism Scalability
Lecture3: Blocking Tiling
Lecture4: Coarsening Tiling
Lecture5: Data Optimization
Lecture6: Input Binning
Lecture7: Input Compaction
Lecture8: Privatization

See also:

http://freevideolectures.com/Course/2880/Advanced-algorithmic-techniques-for-GPUs/1

Massive Parallelism - Regularity



Main Hurdles to Overcome

- Serialization due to conflicting use of critical resources

- Over subscription of Global Memory bandwidth

- Load imbalance among parallel threads

Computational Thinking Skills

- The ability to translate/formulate domain problems into computational models that can be solved efficiently by available computing resources
  - Understanding the relationship between the domain problem and the computational models
  - **Understanding the strength and limitations of the computing devices**
  - **Defining problems and models to enable efficient computational solutions**

DATA ACCESS CONFLICTS

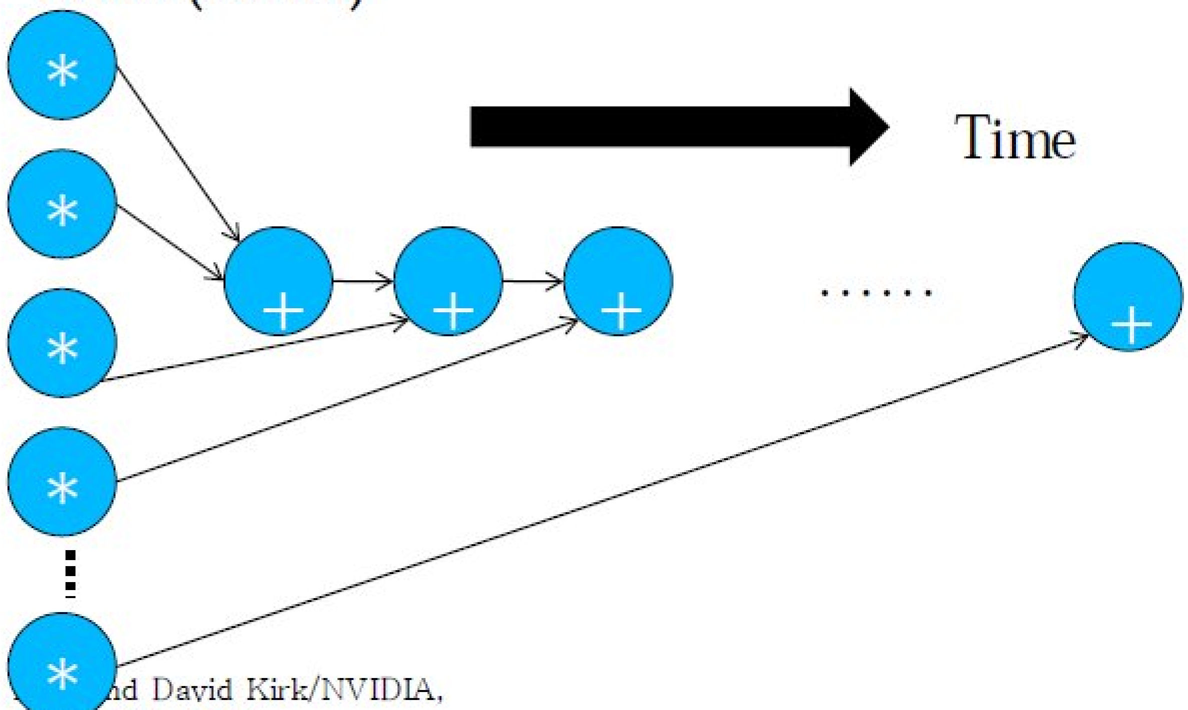
Conflicting Data Accesses Cause Serialization and Delays

- Massively parallel execution cannot afford serialization

- Contentions in accessing critical data causes serialization

A Simple Example

- A naïve inner product algorithm of two vectors of one million elements each
  - All multiplications can be done in time unit (parallel)
  - Additions to a single accumulator in one million time units (serial)



How much can conflicts hurt?

- Amdahl's Law
  - If fraction X of a computation is serialized, the speedup can not be more than 1/(1-X)
- In the previous example, X = 50%
  - Half the calculations are serialized
  - No more than 2X speedup, no matter how many computing cores are used

GLOBAL MEMORY BANDWIDTH

Global Memory Bandwidth

Ideal

Reality

Global Memory Bandwidth

- Many-core processors have limited off-chip memory access bandwidth compared to peak compute throughput
- Fermi
  - 1 TFLOPS SPFP peak throughput
  - 0.5 TFLOPS DPFP peak throughput
  - 144 GB/s peak off-chip memory access bandwidth
    - 36 G SPFP operands per second
    - 18 G DPFP operands per second
  - To achieve peak throughput, a program must perform 1,000/36 = ~28 SPFP (14 DPFP) arithmetic operations for each operand value fetched from off-chip memory

LOAD BALANCE

Load Balance

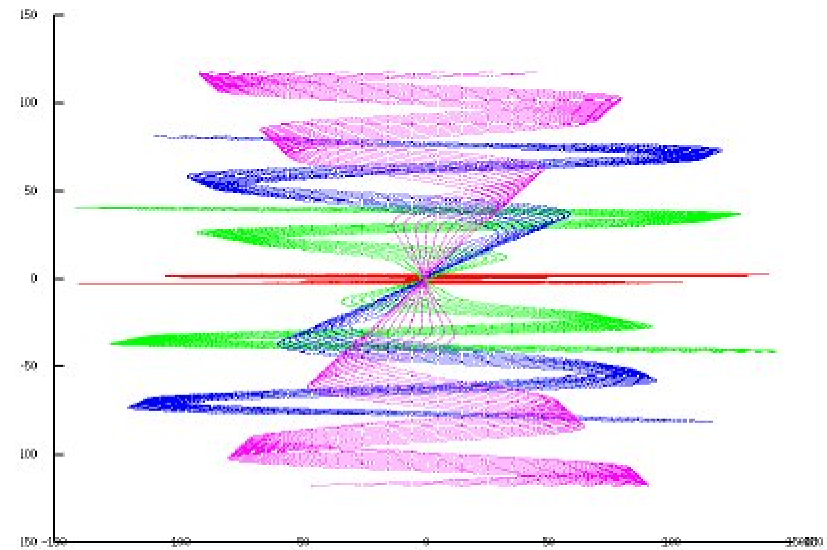
- The total amount of time to complete a parallel job is limited by the thread that takes the longest to finish

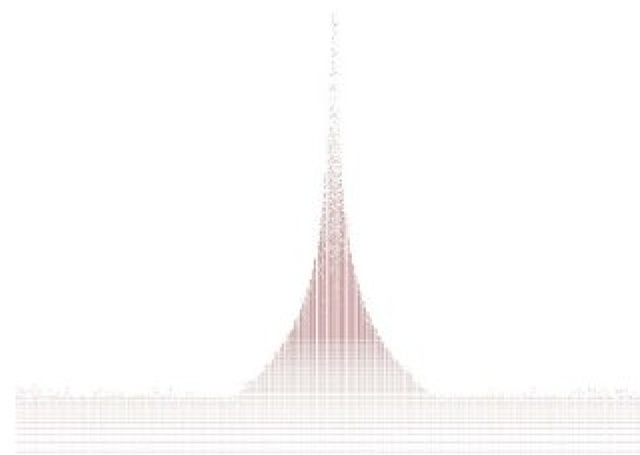
How bad can it be?

- Assume that a job takes 100 units of time for one person to finish
  - If we break up the job into 10 parts of 10 units each and have fo10 people to do it in parallel, we can get a 10X speedup
  - If we break up the job into 50, 10, 5, 5, 5, 5, 5, 5, 5, 5 units, the same 10 people will take 50 units to finish, with 9 of them idling for most of the time. We will get no more than 2X speedup.

How does imbalance come about?

- Non-uniform data distributions
  - Highly concentrated spatial data areas
  - Astronomy, medical imaging, computer vision, rendering, …
- If each thread processes the input data of a given spatial volume unit, some will do a lot more work than others





Eight Algorithmic Techniques (so far)

Technique	Contention	Bandwidth	Locality	Efficiency	Load Imbalance	CPU Leveraging
Tiling		X	X			
Privatization	X		X			
Regularization				X	X	X
Compaction		X				
Binning		X	X	X		X
Data Layout Transformation	X		X			
Thread Coarsening	X	X	X	X		
Scatter to Gather Conversion	X					

http://courses.engr.illinois.edu/ece598/hk/

You can do it.

- Computational thinking is not as hard as you may think it is.
  - Most techniques have been explained, if at all, at the level of computer experts.
  - The purpose of the course is to make them accessible to domain scientists and engineers.

ANY MORE QUESTIONS?