

---

# Emerging Technologies on the Pathway to Extreme Scale Computing

---

Jeffrey Vetter

*Presented to the INT 11-2C Exascale Workshop  
University of Washington*



<http://ft.ornl.gov> ♦ [vetter@computer.org](mailto:vetter@computer.org)

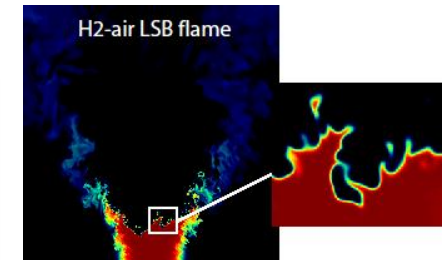
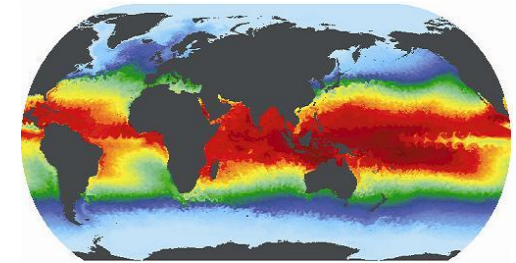
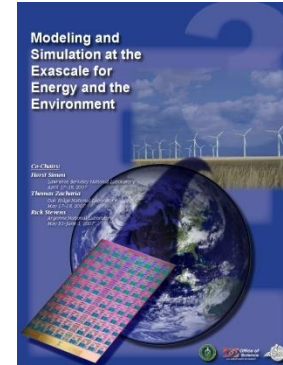
# In a nutshell ...

- Toward Exascale
  - Highlights from recent projections for exascale
- Challenges
  - Micro, macro power
  - Memory capacity and bandwidth
  - Parallelism
  - Programmability
  - Algorithms
- Research solutions
  - Heterogeneity with GPUs: Keeneland
  - Programming models: SHOC, Maestro
  - NVRAM
  - In situ analysis

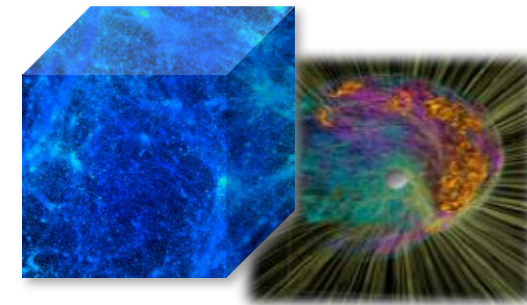
# Toward Exascale

# Process for identifying exascale applications and technology for DOE missions ensures broad community input

- **Town Hall Meetings April-June 2007**
- **Scientific Grand Challenges Workshops Nov, 2008 – Oct, 2009**
  - **Climate Science (11/08),**
  - **High Energy Physics (12/08),**
  - **Nuclear Physics (1/09),**
  - **Fusion Energy (3/09),**
  - **Nuclear Energy (5/09),**
  - **Biology (8/09),**
  - **Material Science and Chemistry (8/09),**
  - **National Security (10/09)**
  - **Cross-cutting technologies (2/10)**
- **Exascale Steering Committee**
  - **“Denver” vendor NDA visits 8/2009**
  - **SC09 vendor feedback meetings**
  - **Extreme Architecture and Technology Workshop 12/2009**
- **International Exascale Software Project**
  - **Santa Fe, NM 4/2009; Paris, France 6/2009; Tsukuba, Japan 10/2009, etc.**



MISSION IMPERATIVES



FUNDAMENTAL SCIENCE

# Holistic View of HPC

Performance, Resilience, Power, Programmability

## Applications

- Materials
- Climate
- Fusion
- National Security
- Combustion
- Nuclear Energy
- Cybersecurity
- Biology
- High Energy Physics
- Energy Storage
- Photovoltaics
- National Competitiveness
- Usage Scenarios
  - Ensembles
  - UQ
  - Visualization
  - Analytics

## Programming Environment

- Domain specific
  - Libraries
  - Frameworks
  - Templates
  - Domain specific languages
  - Patterns
  - Autotuners
- Platform specific
  - Languages
  - Compilers
  - Interpreters/Scripting
  - Performance and Correctness Tools
  - Source code control

## System Software

- Resource Allocation
- Scheduling
- Security
- Communication
- Synchronization
- Filesystems
- Instrumentation
- Virtualization

## Architectures

- Processors
  - Multicore
  - Graphics Processors
  - FPGA
  - DSP
- Memory and Storage
  - Shared (cc, scratchpad)
  - Distributed
  - RAM
  - Storage Class Memory
  - Disk
  - Archival
- Interconnects
  - Infiniband
  - IBM Torrent
  - Cray Gemini, Aires
  - BGL/P/Q
  - 1/10/100 GigE

# Contemporary Systems

Date	System	Location	Comp	Comm	Peak (PF)	Power (MW)
2010	Tianhe-1A	NSC in Tianjin	Intel + NVIDIA	Proprietary	4.7	4.0
2010	Nebulae	NSC In Shenzhen	Intel + NVIDIA	IB	2.9	2.6
2010	Tsubame 2	TiTech	Intel + NVIDIA	IB	2.4	1.4
2011	K Computer (612 cabinets)	Kobe	SPARC64 VIIIfx	Tofu	8.7	9.8
~2012	Cray 'Titan'	ORNL	AMD + NVIDIA	Gemini	20?	7?
~2012	BlueWaters	NCSA/UIUC	POWER7	IBM Hub	10?	10?
~2012	BlueGeneQ	ANL	SoC	IBM	10?	
~2012	BlueGeneQ	LLNL	SoC	IBM	20?	
	Others...					

# Tianhe-1A uses 7000+ NVIDIA GPUs

- Tianhe-1A uses
  - 7,168 NVIDIA Tesla M2050 GPUs
  - 14,336 Intel Westmeres
- Performance
  - 4.7 PF peak
  - 2.5 PF sustained on HPL
- 4.04 MW
  - If Tesla GPU's were not used in the system, the whole machine could have needed 12 megawatts of energy to run with the same performance, which is equivalent to 5000 homes
- Custom fat-tree interconnect
  - 2x bandwidth of Infiniband QDR

The New York Times Business Day  
**Technology**

WORLD U.S. N.Y. / REGION BUSINESS TECHNOLOGY SCIENCE HEALTH SPORTS OPINION

Search Technology   Inside Technology  
[Internet](#) | [Start-Ups](#) | [Business Computing](#) | [Companies](#) **Bits** **Blog**

## China Wrests Supercomputer Title From U.S.

By ASHLEE VANCE  
Published: October 28, 2010

A Chinese scientific research center has built the fastest supercomputer ever made, replacing the United States as maker of the swiftest machine, and giving China bragging rights as a technology superpower.

[Enlarge This Image](#)



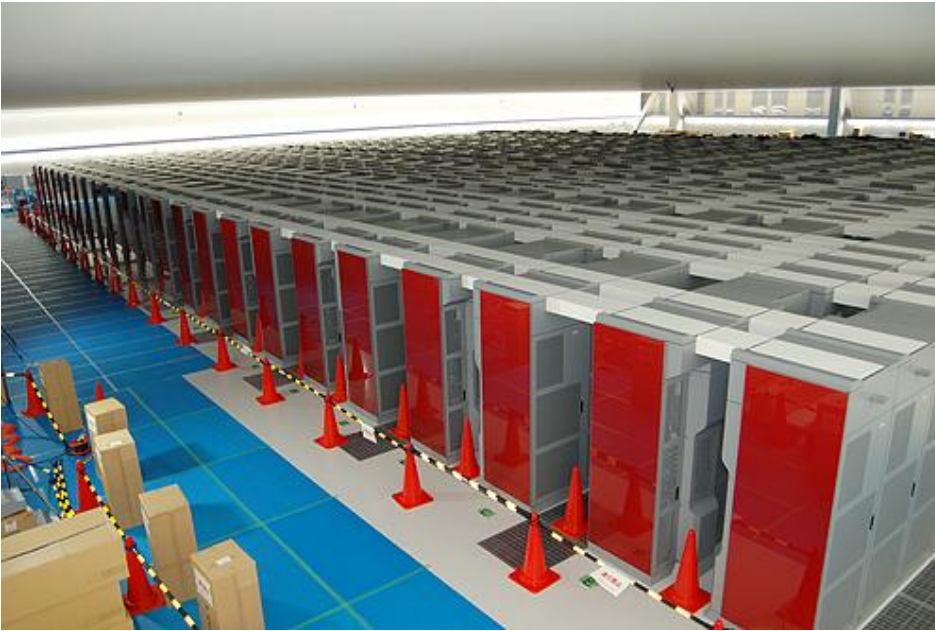
The computer, known as Tianhe-1A, has 1.4 times the horsepower of the current top computer, which is at a national laboratory in Tennessee, as measured by the standard test used to gauge how well the systems handle mathematical calculations, said Jack Dongarra, a [University of Tennessee](#) computer scientist who maintains the official supercomputer rankings.

Although the official list of the top 500 fastest machines, which comes out every six months, is not due to be completed by Mr. Dongarra until next week, he said the Chinese computer “blows away the existing No. 1 machine.” He added, “We don’t close the books until Nov. 1, but I would say it is unlikely we will see a system that is faster.”

Nvidia

- [RECOMMEND](#)
- [TWITTER](#)
- [SIGN IN TO E-MAIL](#)
- [PRINT](#)
- [REPRINTS](#)
- [SHARE](#)

# Recent news - K

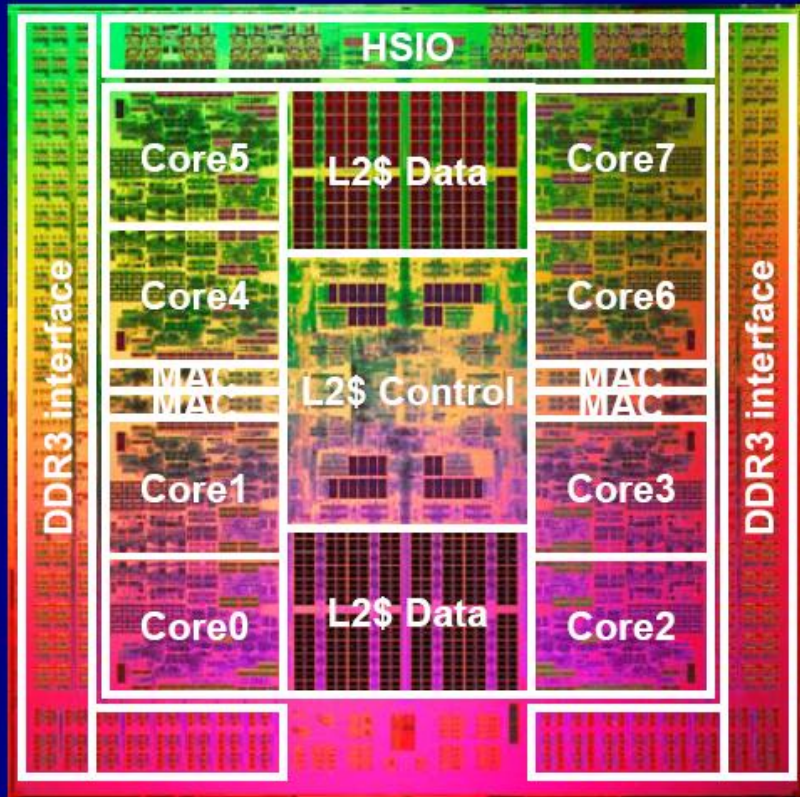


- #1 on TOP500
- 8.162 PF (93% of peak)
  - 3.1x TOP500 #2
  - 9.8 MW
- 672 racks
- 68,544 processors, 1PB memory



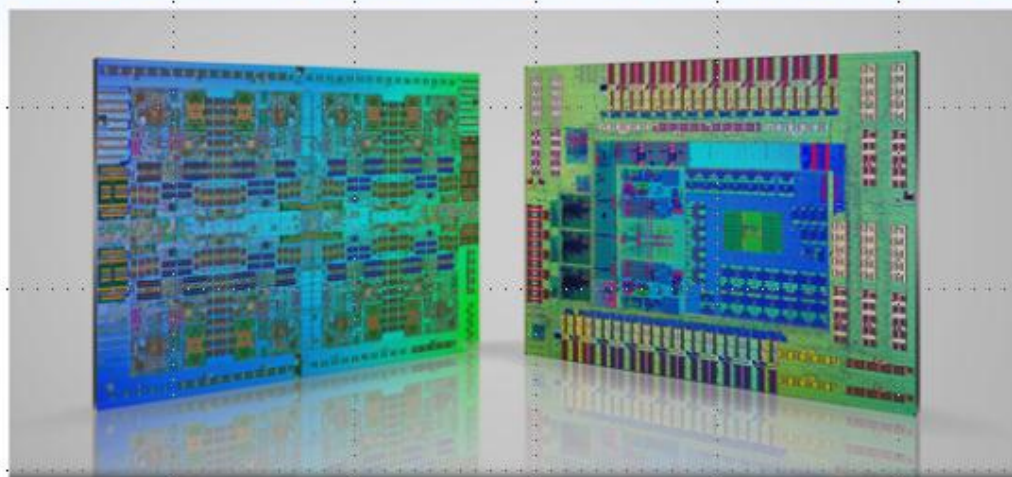


# SPARC64™ VIIIfx Chip Overview



- **Architecture Features**
  - 8 cores
  - Shared 5 MB L2\$
  - Embedded Memory Controller
  - 2 GHz
- **Fujitsu 45nm CMOS**
  - 22.7mm x 22.6mm
  - 760M transistors
  - 1271 signal pins
- **Performance (peak)**
  - 128GFlops
  - 64GB/s memory throughput
- **Power**
  - 58W (TYP, 30°C)
  - Water Cooling – Low leakage power and High reliability

# Heart of Blue Waters: Two New Chips



## Power7 Chip

*Up to 256 GF peak performance*

3.5–4.0+ GHz

Up to 8 cores, 32 threads

Caches

L1 (2x64 KB), L2 (256 KB),

L3 (32 MB)

Memory Subsystem

Two memory controllers:

128 GB/s memory

## PERCS Hub Chip

*1.128 TB/s total bandwidth*

Connections:

192 GB/s QCM connection

896 GB/s to other QCMs

40 GB/s general purpose I/O

## Building Blue Waters

# Blue Waters

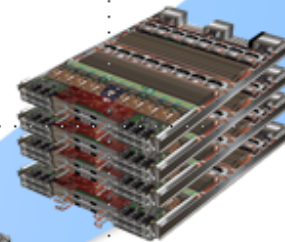
Blue Waters will be the most powerful computer in the world for scientific research when it comes on line in 2011-2.



**Blue Waters**  
≈10 PF Peak  
~1 PF sustained  
≈300,000 cores  
≈1 PB of memory  
>25 PB of disk storage  
500 PB of archival storage  
≥100 Gbps connectivity

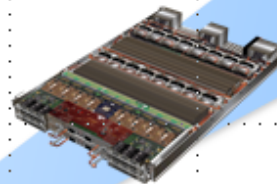
### Blue Waters 3-Rack Building Block

8 IH Supernodes  
256 TF (peak)  
32 TB memory  
128 TB/s memory  
4 Storage systems (>500 TB)  
10 Tape drive connections



### IH Supernode

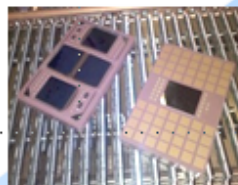
4 IH Server Nodes  
1024 cores  
Up to 32 TF (peak)  
41 TB memory  
16 TB/s  
32 Hub chips  
36 TB/s



### IH Server Node

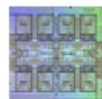
8 QCM's (256 cores)  
Up to 8 TF (peak)  
1 TB memory  
4 TB/s  
8 Hub chips  
9 TB/s  
Power supplies  
PCIe slots

*Fully water cooled*



### Quad-chip Module

4 Power7 chips  
Up to 1 TF (peak)  
128 GB memory  
512 GB/s



### Power7 Chip

8 cores, 32 threads  
L1, L2, L3 cache (32 MB)  
Up to 256 GF (peak)  
128 Gb/s memory bw  
45 nm technology

### Hub Chip

1.128 TB/s

Blue Waters is built from components that can also be used to build systems with a wide range of capabilities—from desktside to beyond Blue Waters.

# **Exascale Trends and Challenges**

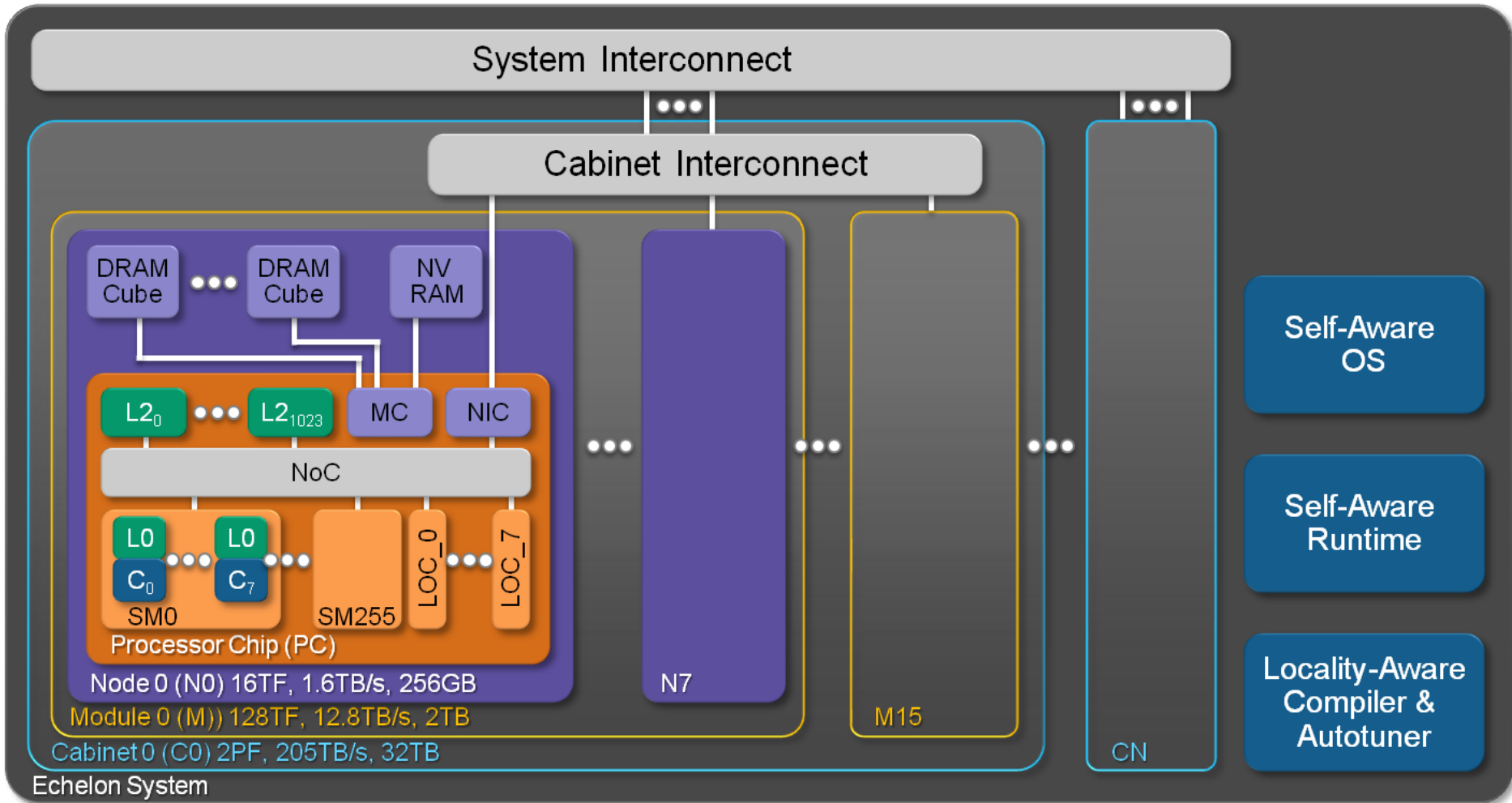
# Notional Exascale Architecture Targets

System attributes	2001	2010	"2015"		"2018"	
System peak	10 Tera	2 Peta	200 Petaflop/sec		1 Exaflop/sec	
Power		6 MW	15 MW		20 MW	
System memory	0.006 PB	0.3 PB	5 PB		32-64 PB	
Node performance	0.024 TF	0.125 TF	0.5 TF	7 TF	1 TF	10 TF
Node memory BW		25 GB/s	0.1 TB/sec	1 TB/sec	0.4 TB/sec	4 TB/sec
Node concurrency	16	12	O(100)	O(1,000)	O(1,000)	O(10,000)
System size (nodes)	416	18,700	50,000	5,000	1,000,000	100,000
Total Node Interconnect BW		1.5 GB/s	150 GB/sec	1 TB/sec	250 GB/sec	2 TB/sec
MTTI		day	O(1 day)		O(1 day)	

# Note the Uneven Impact on System Balance!

	2010	2018	Factor Change
System peak	2 Pf/s	1 Ef/s	500
Power	6 MW	20 MW	3
System Memory	0.3 PB	10 PB	33
Node Performance	0.125 Tf/s	10 Tf/s	80
Node Memory BW	25 GB/s	400 GB/s	16
Node Concurrency	12 cpus	1,000 cpus	83
Interconnect BW	1.5 GB/s	50 GB/s	33
System Size (nodes)	20 K nodes	1 M nodes	50
Total Concurrency	225 K	1 B	4,444
Storage	15 PB	300 PB	20
Input/Output bandwidth	0.2 TB/s	20 TB/s	100

# NVIDIA Echelon System Sketch



DARPA Echelon team: NVIDIA, ORNL, Micron, Cray, Georgia Tech, Stanford, UC-Berkeley, U Penn, Utah, Tennessee, Lockheed Martin

# Challenges to Exascale

## Performance Growth

- 1) **System power** is the primary constraint
- 2) **Memory** bandwidth and capacity are not keeping pace
- 3) **Concurrency** (1000x today)
- 4) **Processor** architecture is an open question
- 5) **Programming model** heroic compilers will not hide this
- 6) **Algorithms** need to minimize data movement, not flops
- 7) **I/O bandwidth** unlikely to keep pace with machine speed
- 8) **Reliability and resiliency** will be critical at this scale
- 9) **Bisection bandwidth** limited by cost and energy

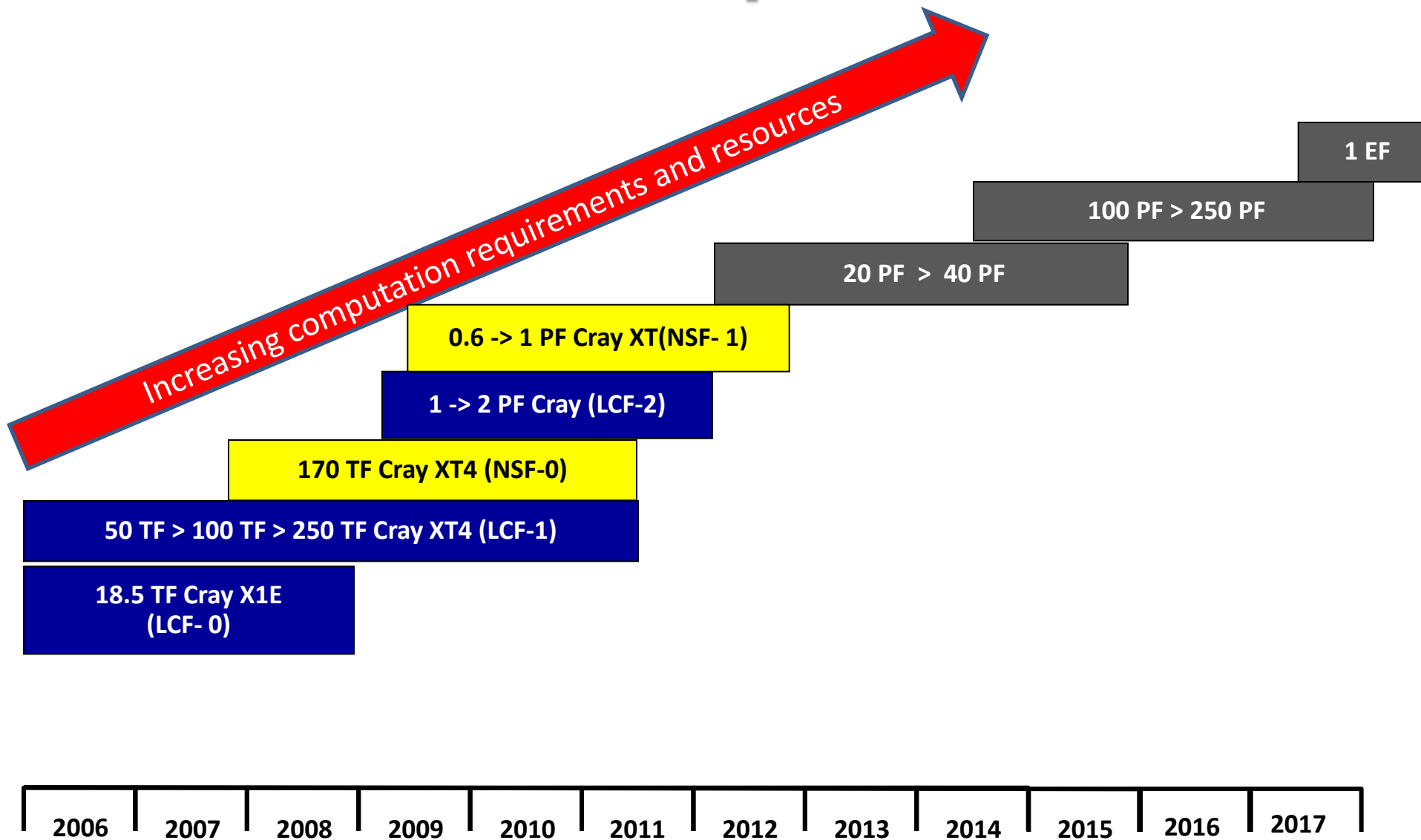
*Unlike the last 20 years most of these (1-7) are equally important across scales, e.g., 100 10-PF machines*



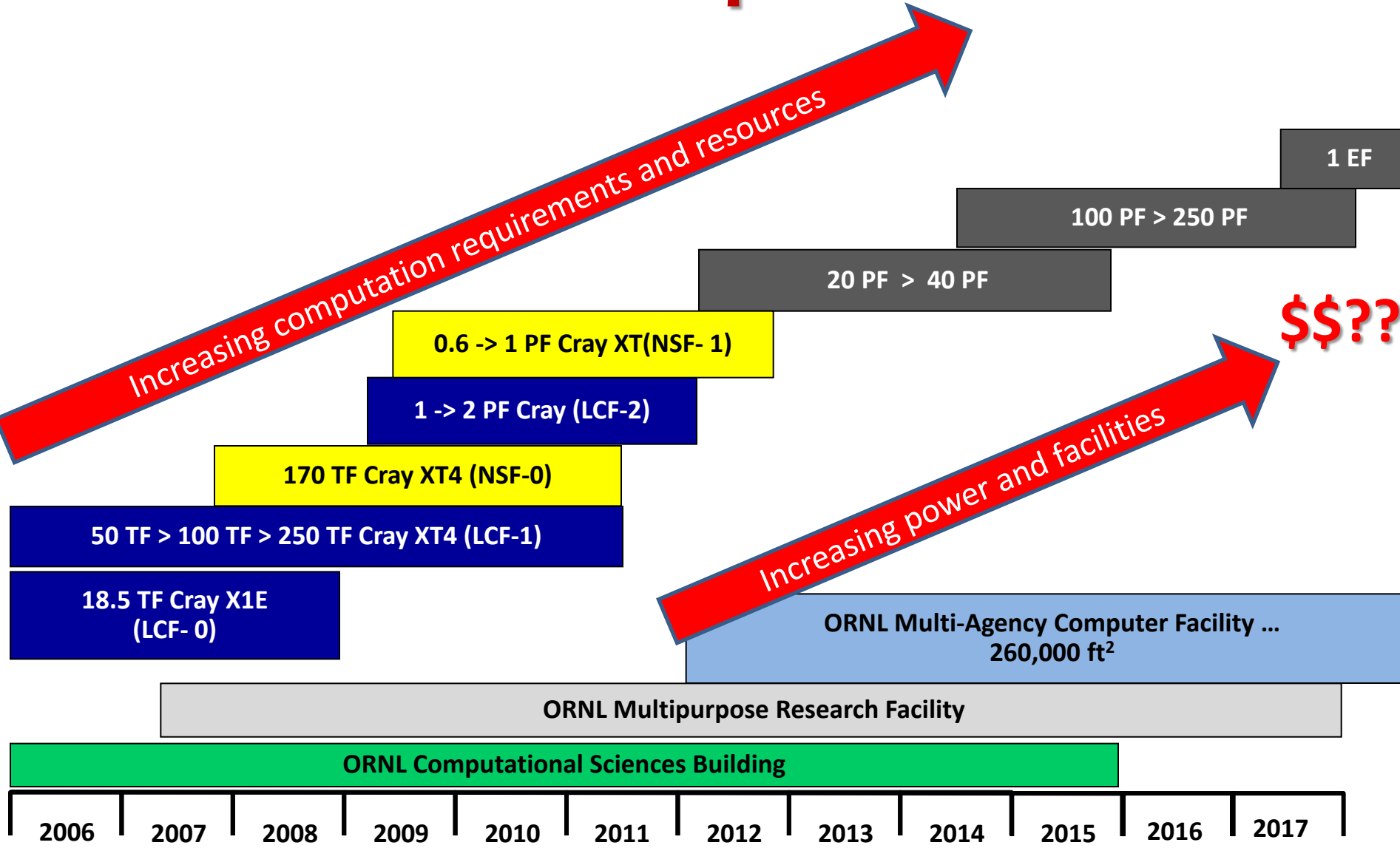
# #1: Power

Both macro and micro energy trends drive all other factors

# ORNL Roadmap to Exascale



# ORNL Roadmap to Exascale



If energy costs ~\$1/MW/yr, then how much is the energy cost for an exascale system?!?!?

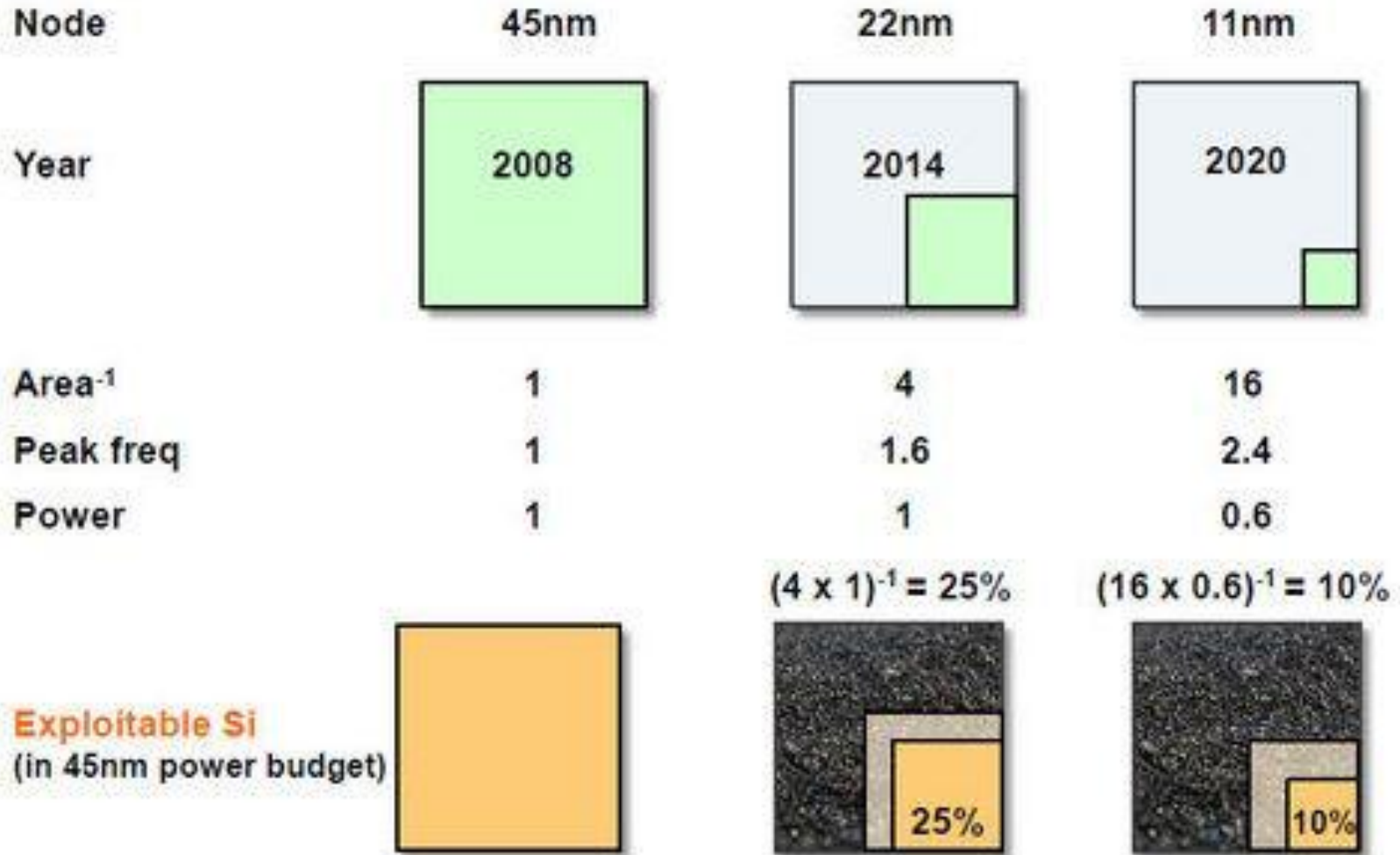
# Facilities and Power ... Not just ORNL



**... but the more important trend is power management at the chip level ...**

**#3/4: (Uncertainty in)  
Concurrency and Processor  
Architecture**

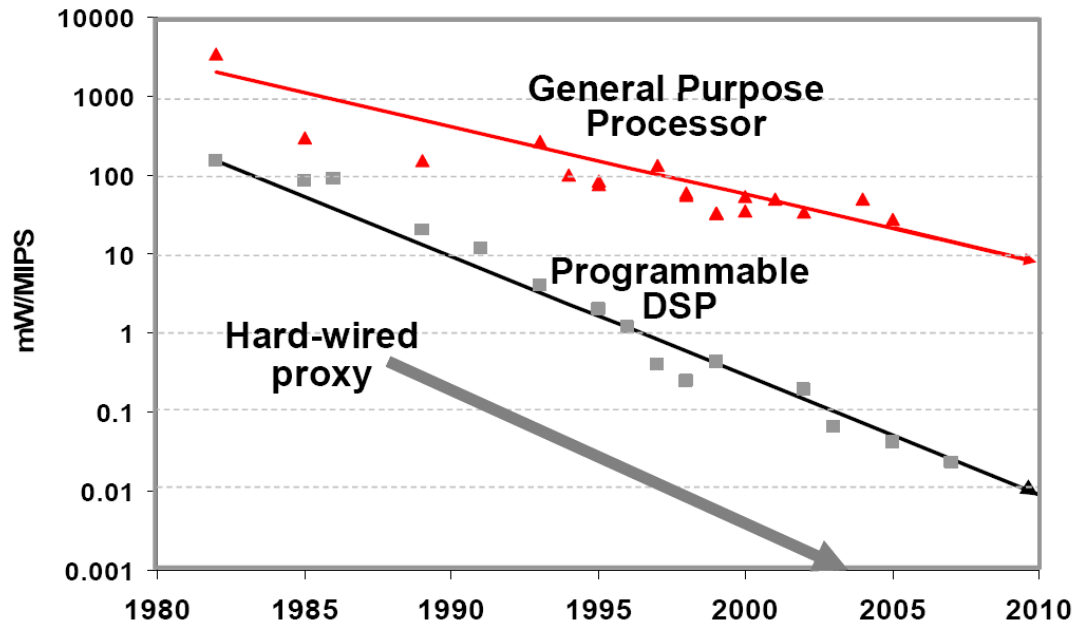
# Dark Silicon



Source: ITRS 2008

# Mobile and Embedded Systems Community has Experiences

No single architecture solves all power problems



- Industry has debated merits of each architecture for decades...
- Combination of all approaches optimizes power and performance



# CPU GPU SoC: Features & Block Diagram

## CPU

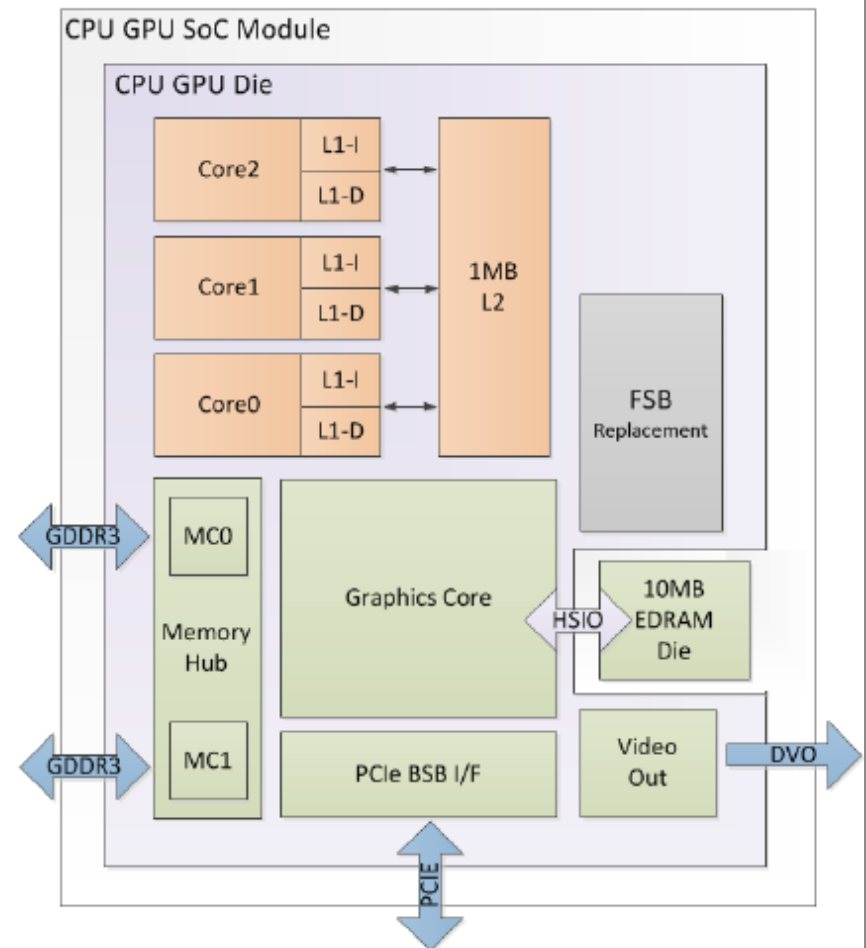
- Three 3.2 GHz PowerPC® cores
- Shared 1MB L2 cache
- Per Core:
  - Dual Thread Execution
  - 32K L1 I-cache, 32K L1 D-cache
  - 2-issue per cycle
  - Branch, Integer, Load/Store Units
  - VMX128 Units enhanced for games

## GPU

- 48 parallel unified shaders
- 24 billion shader instructions per second
- 4 billion pixels/sec pixel fill rate
- 500 million triangles/sec geometry rate
- High Speed IO interface to 10 MB EDRAM

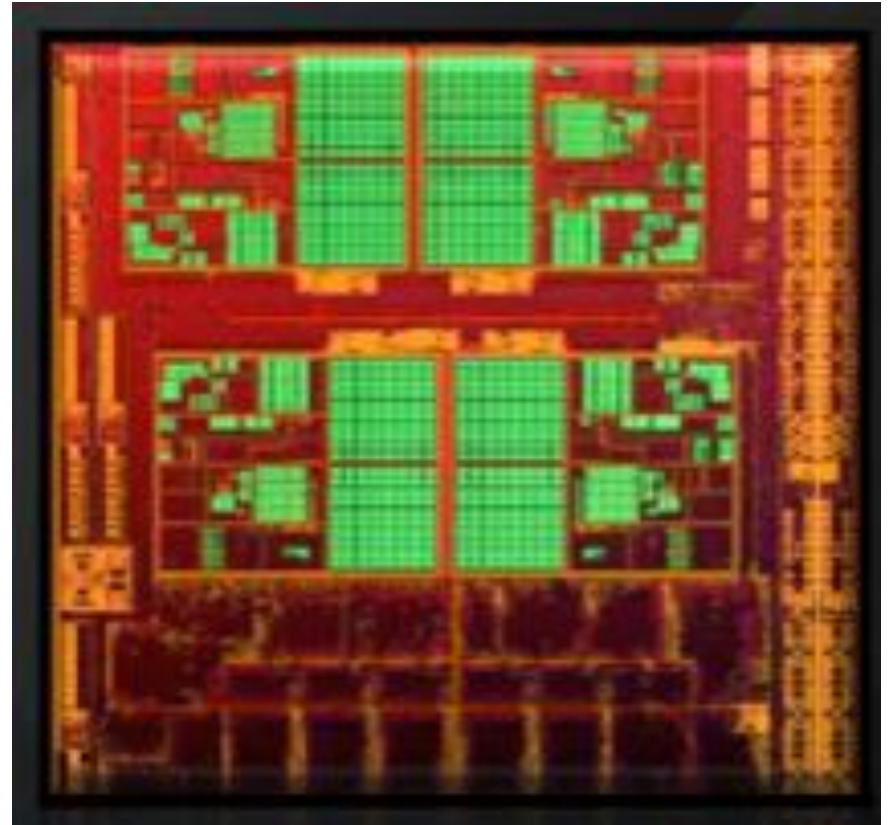
## Compatibility

- Functional and Performance equivalent to prior Xbox 360 GPU/CPU
- FSB Latency and BW match prior FSB



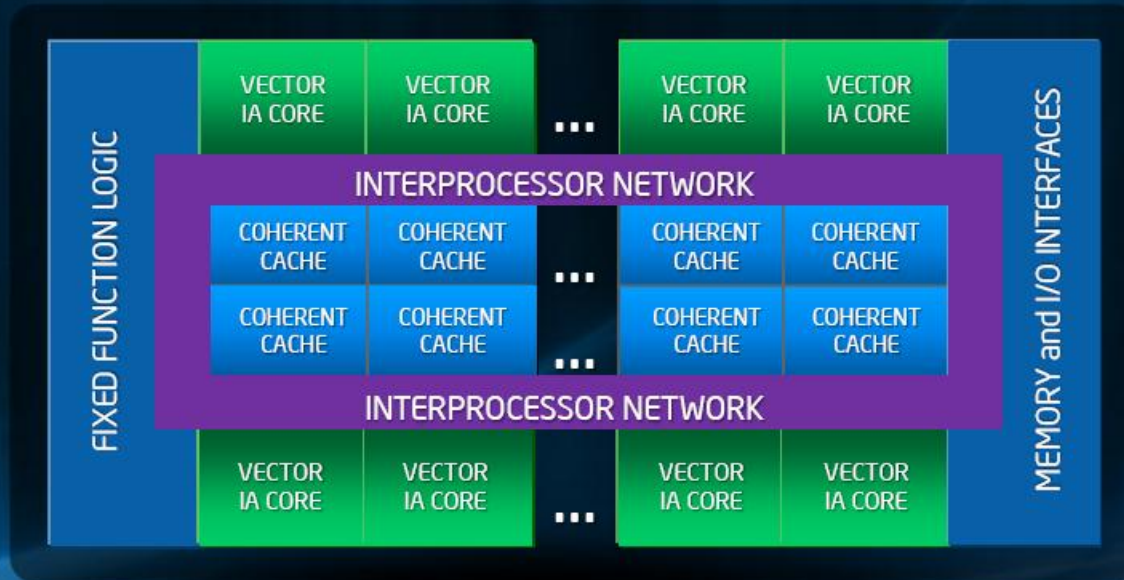
# AMD's Llano: A-Series APU

- Combines
  - 4 x86 cores
  - Array of Radeon cores
  - Multimedia accelerators
  - Dual channel DDR3
- 32nm
- Up to 29 GB/s memory bandwidth
- Up to 500 Gflops SP
- 45W TDP



Source: AMD

# Intel® MIC Architecture: *An Intel Co-Processor Architecture*

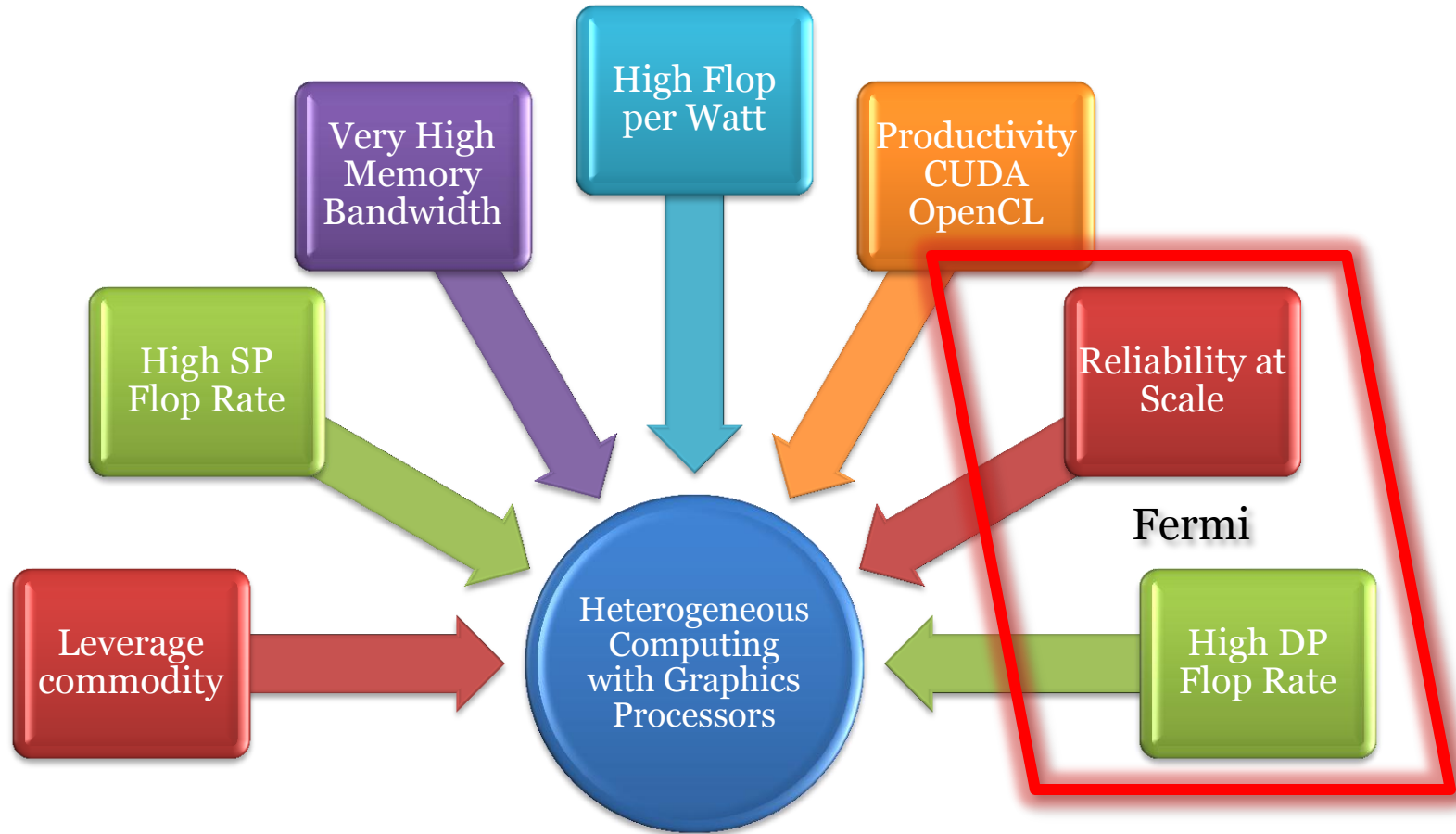


Many cores and many, many more threads  
Standard IA programming and memory model



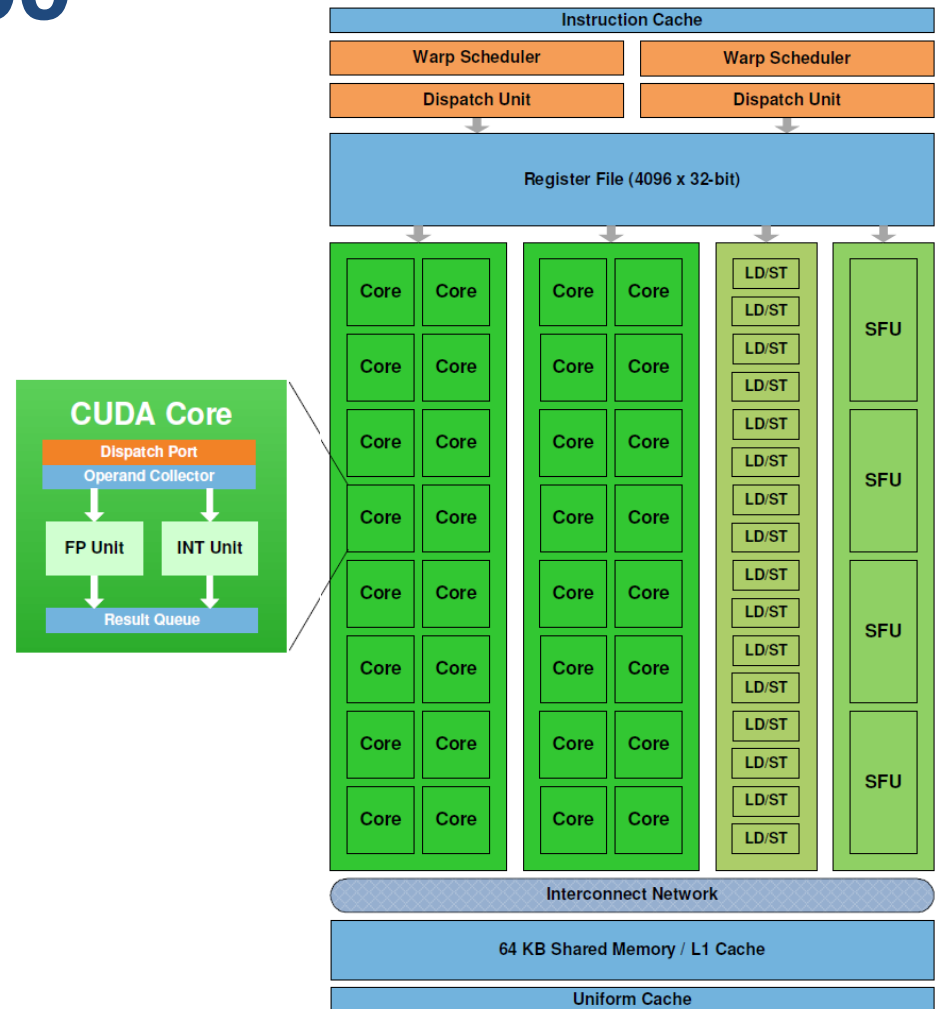
# Graphics Processors

# GPU Rationale – What's different now?

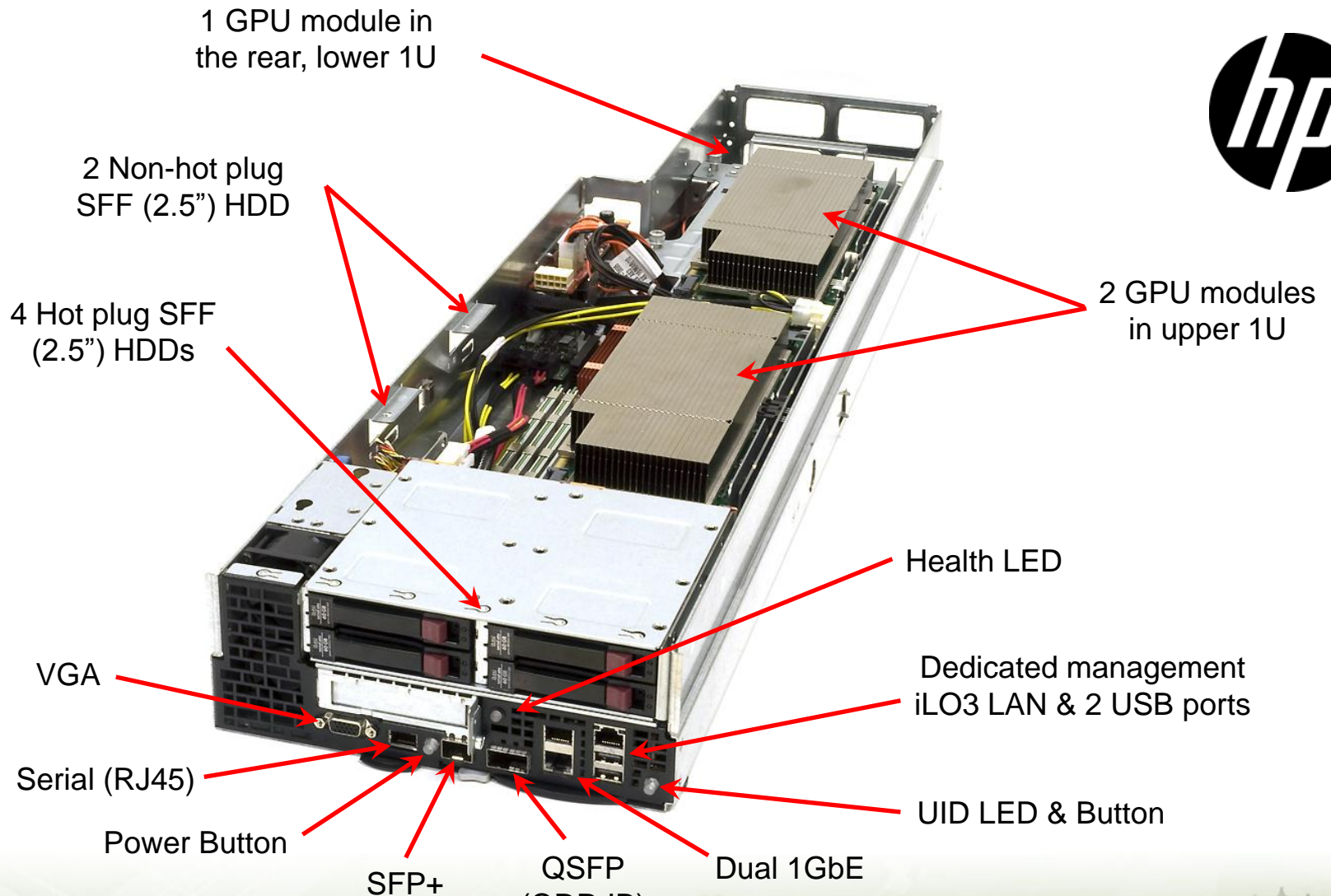


# NVIDIA Fermi/GF100

- 3B transistors in 40nm
- Up to 512 CUDA Cores
  - New IEEE 754-2008 floating-point standard
    - FMA
    - 8× the peak double precision arithmetic performance over NVIDIA's last generation GPU
  - 32 cores per SM, 21k threads per chip
- 384b GDDR5, 6 GB capacity
  - ~120-144 GB/s memory BW
- C/M2070
  - 515 GigaFLOPS DP, 6GB
  - ECC Register files, L1/L2 caches, shared memory and DRAM



# HP ProLiant SL390s G7 2U half width tray



# Keeneland – Initial Delivery System Architecture

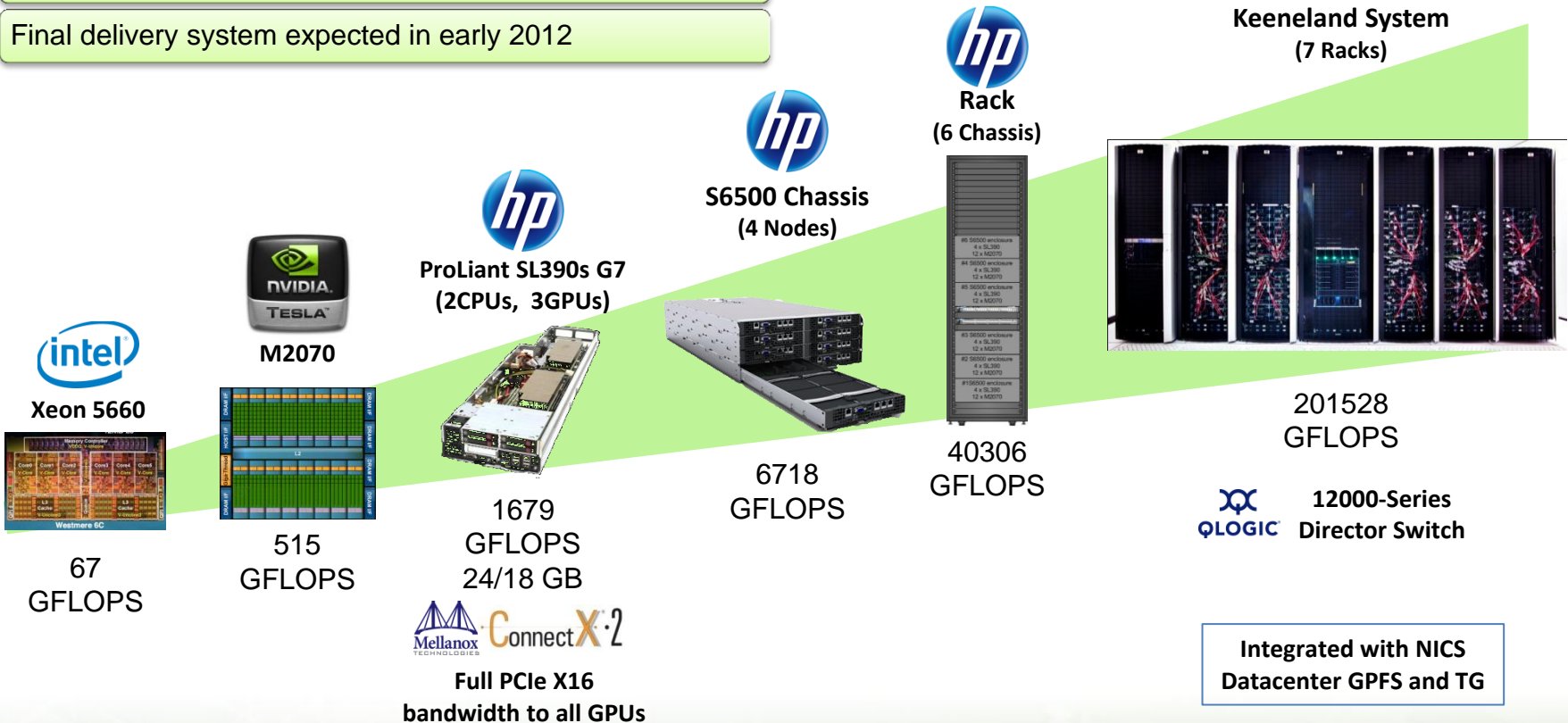


Initial Delivery system procured and installed in Oct 2010

201 TFLOPS in 7 racks (90 sq ft incl service area)

677 MFLOPS per watt on HPL

Final delivery system expected in early 2012





# Keeneland ID installation – 10/29/10



# Organizations using Keeneland

Allinea  
Arizona  
Brown University  
Dartmouth  
Florida State  
George Washington  
Georgia Tech  
HP  
Illinois  
NCAR  
NCSA  
North Carolina  
Northwestern  
NVIDIA  
Ohio State  
ORNL  
OSU  
Purdue

Reservoir Labs  
Rice Univ.  
San Diego  
Stanford  
Temple  
Trideum Corp  
U. Chicago  
U. Cincinnati  
U. Delaware  
U. Florida  
U. Georgia  
U. Illinois  
U. Maryland  
U. New Hampshire  
U. of Oregon  
U. South Calif.  
Univ of Calif, Davis  
UCSD

Univ South Calif.  
Univ of Delaware  
Univ of Calif  
Univ of California, Los Angeles  
Univ of Utah  
Univ of Rochester  
Univ of Wisconsin  
UTK  
Vanderbilt  
Yale

*New users being added weekly*

# Early (Co-design) Success Stories

## Computational Materials

- Quantum Monte Carlo
  - High-temperature superconductivity and other materials science
  - 2008 Gordon Bell Prize
- GPU acceleration speedup of 19x in main QMC Update routine
  - Single precision for CPU and GPU: target single-precision only cards
- Full parallel app is 5x faster, start to finish, on a GPU-enabled cluster on Tesla T10

GPU study: J.S. Meredith, G. Alvarez, T.A. Maier, T.C. Schulthess, J.S. Vetter, "Accuracy and Performance of Graphics Processors: A Quantum Monte Carlo Application Case Study", *Parallel Comput.*, 35(3):151-63, 2009.

Accuracy study: G. Alvarez, M.S. Summers, D.E. Maxwell, M. Eisenbach, J.S. Meredith, J. M. Larkin, J. Levesque, T. A. Maier, P.R.C. Kent, E.F. D'Azevedo, T.C. Schulthess, "New algorithm to enable 400+ TFlop/s sustained performance in simulations of disorder effects in high-Tc superconductors", SuperComputing, 2008. [Gordon Bell Prize winner]

## Combustion

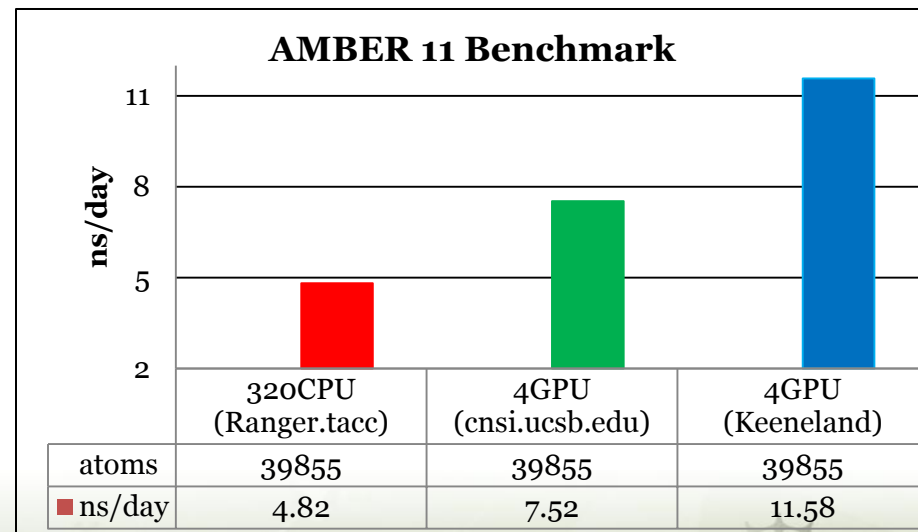
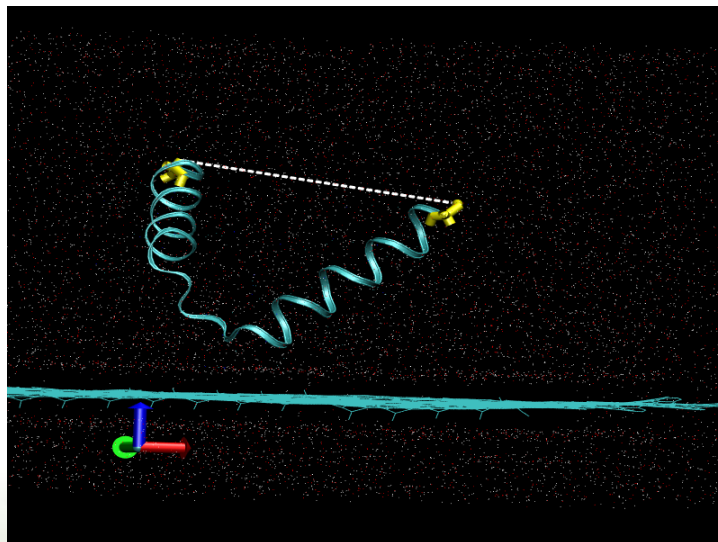
- S3D
  - Massively parallel direct numerical solver (DNS) for the full compressible Navier-Stokes, total energy, species and mass continuity equations
  - Coupled with detailed chemistry
  - Scales to 150k cores on Jaguar
- Accelerated version of S3D's Getrates kernel in CUDA on Tesla T10
  - 31.4x SP speedup
  - 16.2x DP speedup

K. Spafford, J. Meredith, J. S. Vetter, J. Chen, R. Grout, and R. Sankaran. Accelerating S3D: A GPGPU Case Study. Proceedings of the Seventh International Workshop on Algorithms, Models, and Tools for Parallel Computing on Heterogeneous Platforms (HeteroPar 2009) Delft, The Netherlands.

# Peptide folding on surfaces

- Peptide folding on a hydrophobic surface
  - [www.chem.ucsb.edu/~sheagroup](http://www.chem.ucsb.edu/~sheagroup)
- Surfaces can modulate the folding and aggregation pathways of proteins. Here, we investigate the folding of a small helical peptide in the presence of a hydrophobic surface of graphite. Simulations are performed using explicit solvent and a fully atomic representation of the peptide and the surface.

- Benefits of running on a GPU cluster:
  - Reduction in the the number of computing nodes needed: one GPU is at least faster than 8 CPUs in GPU-accelerated AMBER Molecular Dynamics.
  - The large simulations that we are currently running would be prohibitive using CPUs. The efficiency of the CPU parallelization becomes poorer with increasing number of CPUs.
  - It can also decrease consumption of memory and network bandwidth in simulations with large number of atoms.

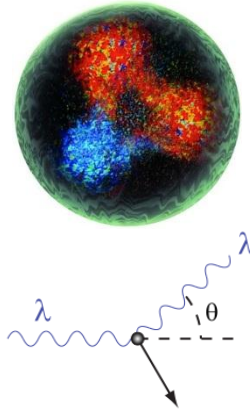


# Hadron Polarizability in Lattice QCD

Andrei Alexandru  
The George Washington University



Understanding the structure of subnuclear particles represents the main challenge for today's nuclear physics. Photons are used to probe this structure in experiments carried out at laboratories around the world. To interpret the results of these experiments we need to understand how electromagnetic field interacts with subnuclear particles. Theoretically, the structure of subnuclear particles is described by Quantum Chromodynamics (QCD). Lattice QCD is a 4-dimensional discretized version of this theory that can be solved numerically. The focus of our project is to understand how the electric field deforms neutrons and protons by computing the polarizability using lattice QCD techniques.

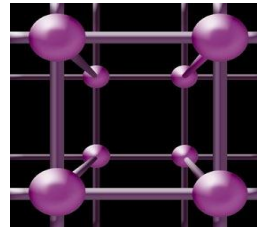
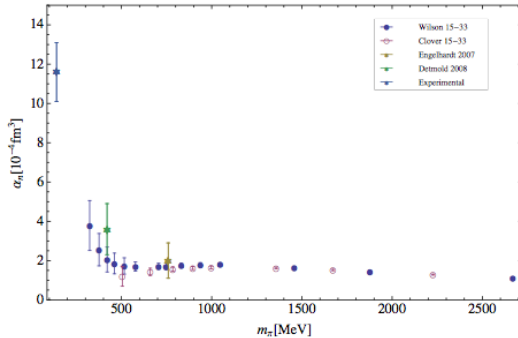


## Why GPUs?

- Lattice QCD simulations require very large bandwidth to run efficiently. GPUs have 10–15 times larger memory bandwidth compared to CPUs.
- Lattice QCD simulations can be efficiently parallelized.
  - Bulk of calculation spent on one kernel.
  - The kernel requires only nearest neighbor information.
  - Cut the lattice into equal sub-lattices. Effectively use single instruction multiple-data (SIMD) paradigm.

[http://samurai.phys.gwu.edu/wiki/index.php/Hadron\\_polarizability](http://samurai.phys.gwu.edu/wiki/index.php/Hadron_polarizability)

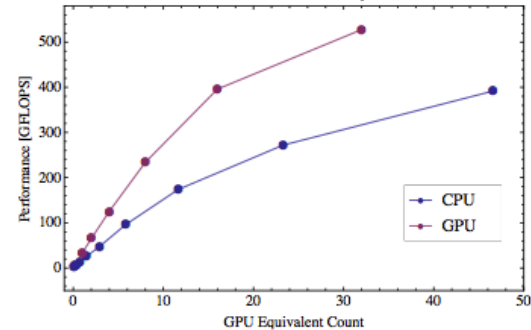
Neutron electric polarizability



Experimental and current values for neutron electric polarizability in lattice QCD.

Alexandru and F. X. Lee, [arXiv:0810.2833]

CPU GPU Performance Comparison



Performance comparison between Keeneland's GPU cluster and Kraken's Cray XT-5 machine. The CPU core count is translated to GPU equivalent count by dividing the total number of CPUs by 22, which is the number of CPU cores equivalent to a single-GPU performance.

A. Alexandru. *et. al*, [arXiv:1103.5103]

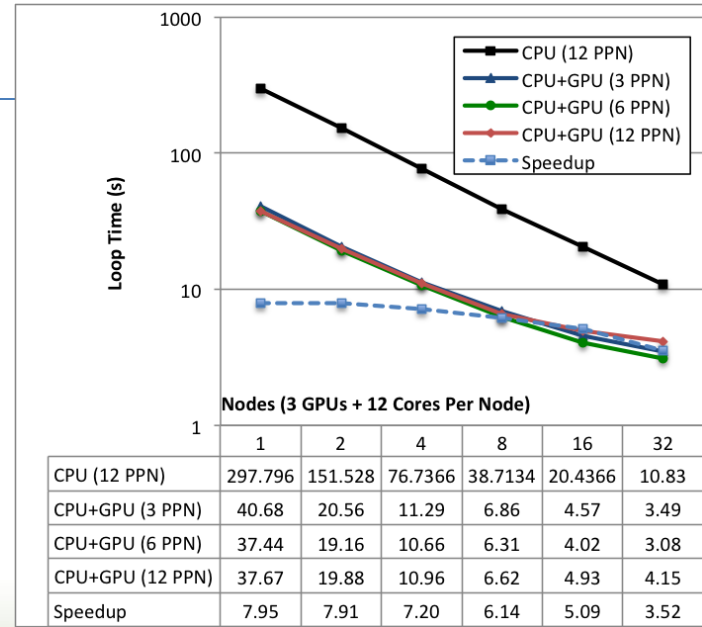
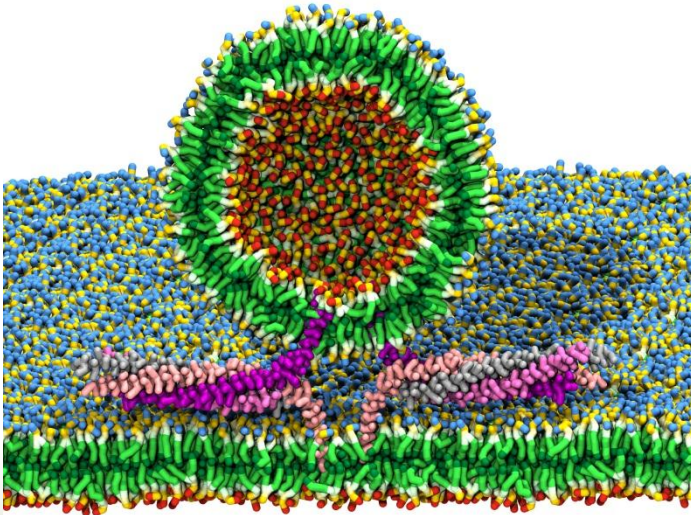


# LAMMPS with GPUs

PI: Axel Kohlmeyer,  
Temple University

- Parallel Molecular Dynamics
  - <http://lammps.sandia.gov>
  - Classical Molecular Dynamics
  - Atomic models, Polymers, Metals, Bio-simulations, Coarse-grain (picture), Ellipsoids, etc.
  - Already good strong and weak scaling on CPUs via MPI

- Better performance on fewer nodes => larger problems faster
- Neighbor, non-bonded force, and long-range GPU acceleration
- Allows for CPU/GPU concurrency
- Implementation and benchmarks by W. Michael Brown, NCCS, ORNL



# **#5: Programming Systems**

# Holistic View of HPC

Performance, Resilience, Power, Programmability

## Applications

- Materials
- Climate
- Fusion
- National Security
- Combustion
- Nuclear Energy
- Cybersecurity
- Biology
- High Energy Physics
- Energy Storage
- Photovoltaics
- National Competitiveness
- Usage Scenarios
  - Ensembles
  - UQ
  - Visualization
  - Analytics

## Programming Environment

- Domain specific
  - Libraries
  - Frameworks
  - Templates
  - Domain specific languages
  - Patterns
  - Autotuners
- Platform specific
  - Languages
  - Compilers
  - Interpreters/Scripting
  - Performance and Correctness Tools
  - Source code control

**Critical**

## System Software

- Resource Allocation
- Scheduling
- Security
- Communication
- Synchronization
- Filesystems
- Instrumentation
- Virtualization

## Architectures

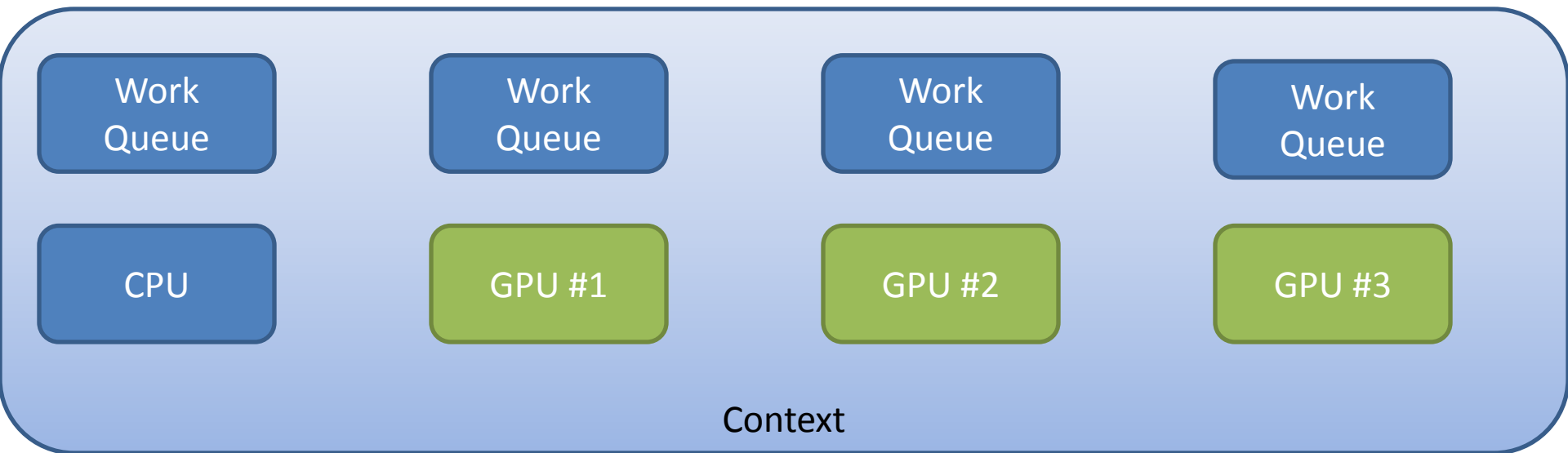
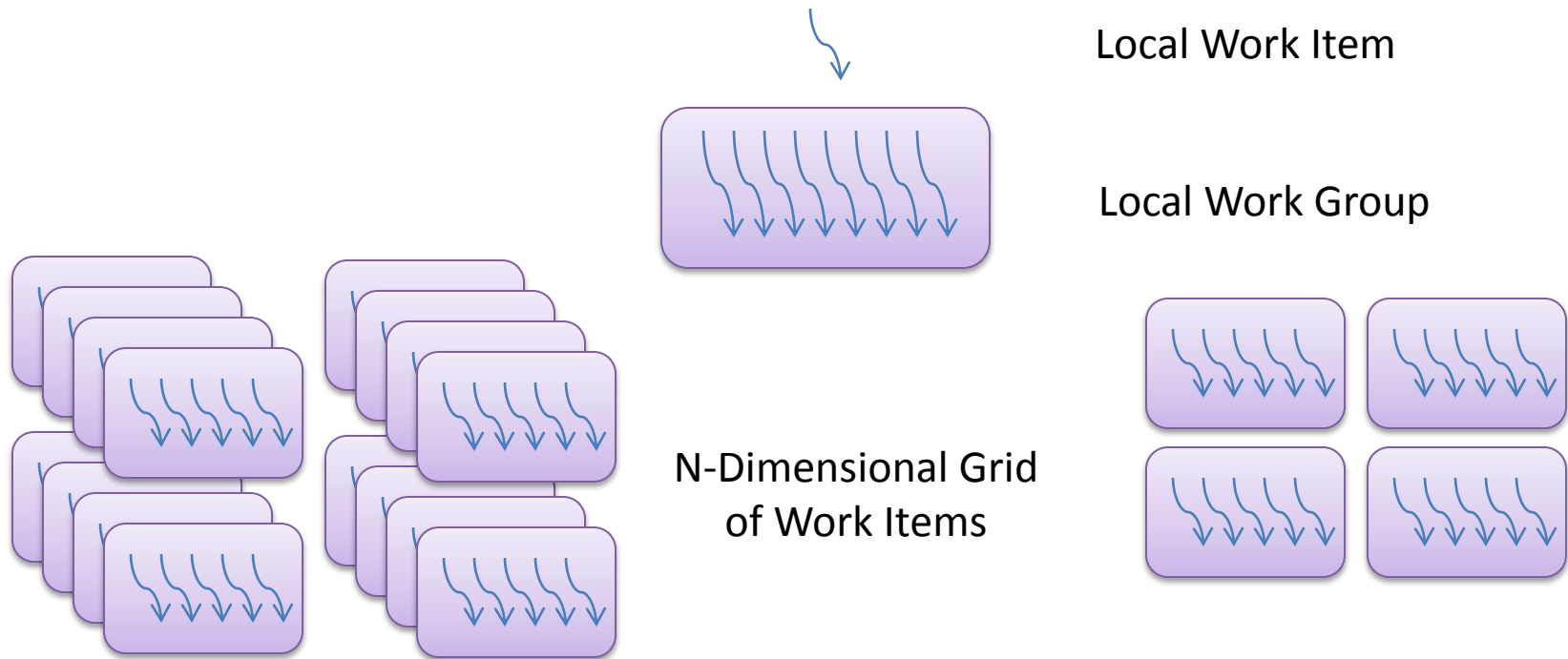
- Processors
  - Multicore
  - Graphics Processors
  - FPGA
  - DSP
- Memory and Storage
  - Shared (cc, scratchpad)
  - Distributed
  - RAM
  - Storage Class Memory
  - Disk
  - Archival
- Interconnects
  - Infiniband
  - IBM Torrent
  - Cray Gemini, Aires
  - BGL/P/Q
  - 1/10/100 GigE



# Keeneland Software Environment

- Integrated with NSF TeraGrid/XD
  - Including TG and NICS software stack
- Programming environments
  - CUDA
  - OpenCL
  - Compilers
    - GPU-enabled
  - Scalable debuggers
  - Performance tools
  - Libraries
- Tools and programming options are changing rapidly
  - HMPP, PGI, LLVM, Polaris, R-stream,
- Additional software activities
  - Performance and correctness tools
  - Scientific libraries
  - Virtualization

# OpenCL Programming Model



# Scalable Heterogeneous Computing (SHOC) Benchmark Suite

- Benchmark suite with a focus on scientific computing workloads, including common kernels like SGEMM, FFT, Stencils
- Parallelized with MPI, with support for multi-GPU and cluster scale comparisons
- Implemented in CUDA and OpenCL for a 1:1 performance comparison
- Includes stability tests
- Performance portability

- Level 0
  - **BusSpeedDownload**: measures bandwidth of transferring data across the PCIe bus to a device.
  - **BusSpeedReadback**: measures bandwidth of reading data back from a device.
  - **DeviceMemory**: measures bandwidth of memory accesses to various type of device memory including global, local, and image memories.
  - **KernelCompile**: measures compile time for several OpenCL kernels, which range in complexity
  - **PeakFlops**: measures maximum achievable floating point performance using a combination of auto-generated and hand-coded kernels.
  - **QueueDelay**: measures the overhead of enqueueing an OpenCL command queue.
- Level 1
  - **FFT**: forward and reverse 1D FFT.
  - **MD**: computation of the Lennard-Jones potential from molecular dynamics, a specific case of the nbody problem.
  - **Reduction**: reduction operation on an array of single precision floating point values.
  - **SGEMM**: single-precision matrix-matrix multiply.
  - **Sort**: scan (also known as parallel prefix sum) on an array of single precision floating point values.
  - **Sort**: sorts an array of key-value pairs using a radix sort algorithm
  - **Stencil2D**: a 9-point stencil operation applied to a 2D data set. In the MPI version, data is distributed across MPI processes organized in a 2D Cartesian topology, with periodic halo exchanges.
  - **Triad**: STREAM Triad operations, implemented in OpenCL.

SHOC 1.0 available  
Downloaded over 1000 times

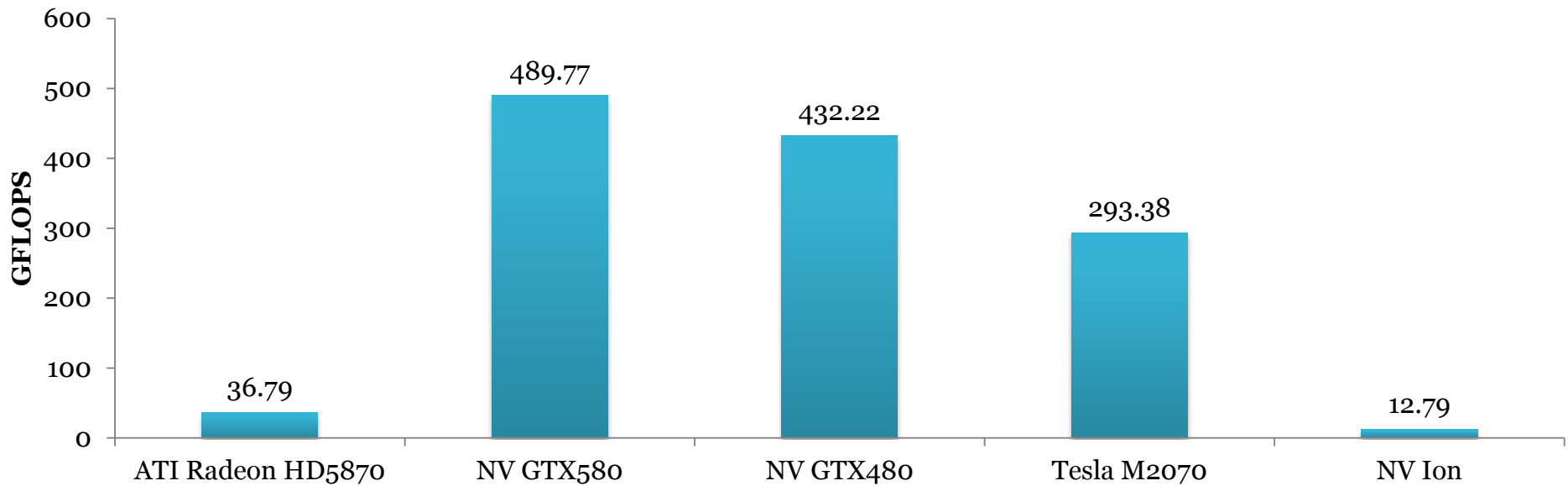
Software available at <http://bit.ly/shocmarx>

A. Danalis, G. Marin, C. McCurdy, J. Meredith, P.C. Roth, K. Spafford, V. Tipparaju, and J.S. Vetter, "The Scalable Heterogeneous Computing (SHOC) Benchmark Suite," in Third Workshop on General-Purpose Computation on Graphics Processors (GPGPU 2010). Pittsburgh, 2010

### Single Precision MD

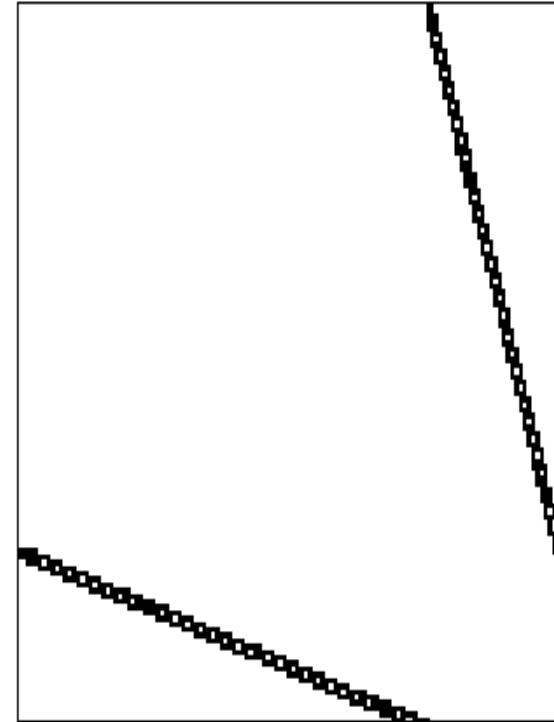


### Single Precision FFT

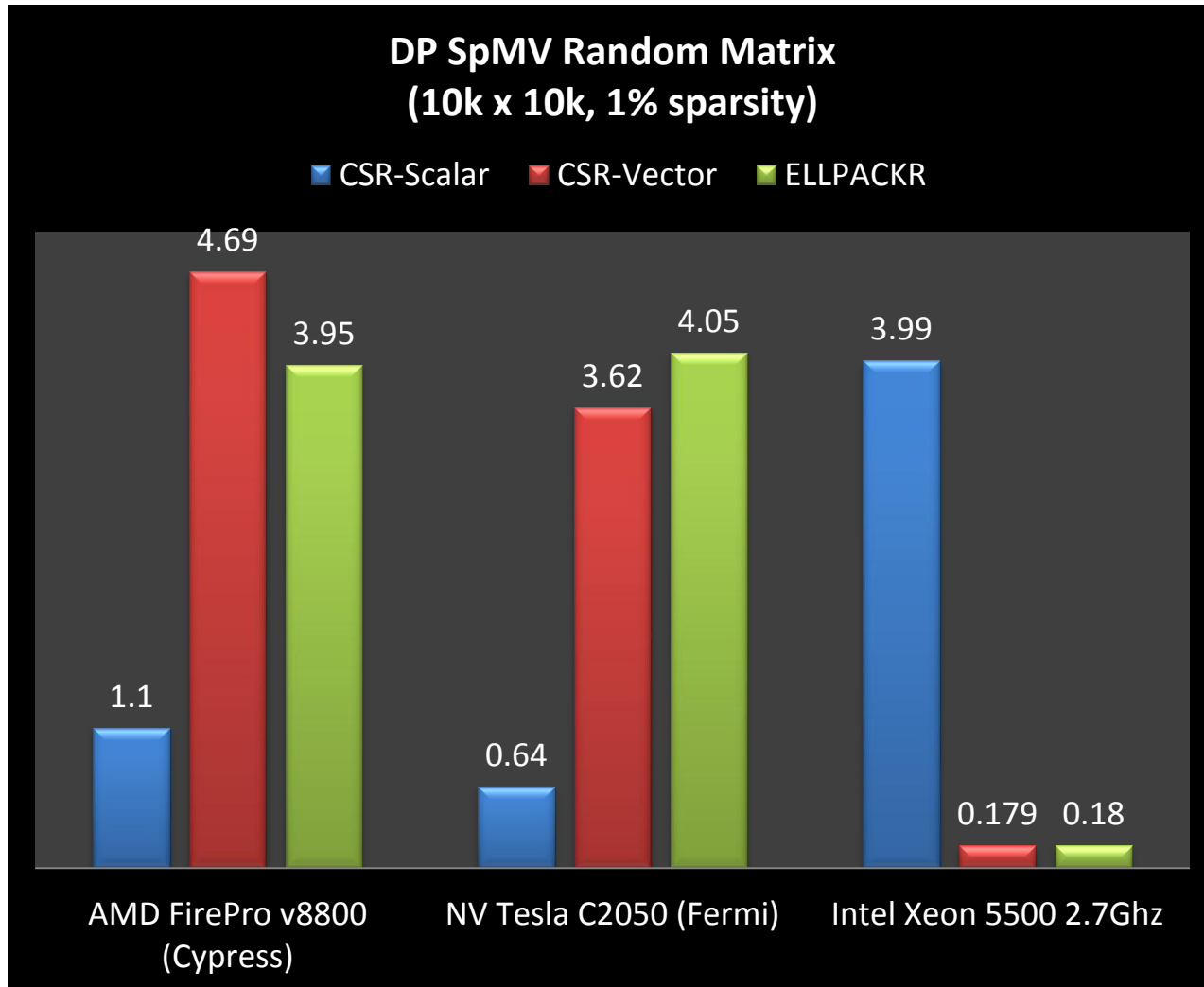


# Example: Sparse Matrix Vector Multiplication (SpMV)

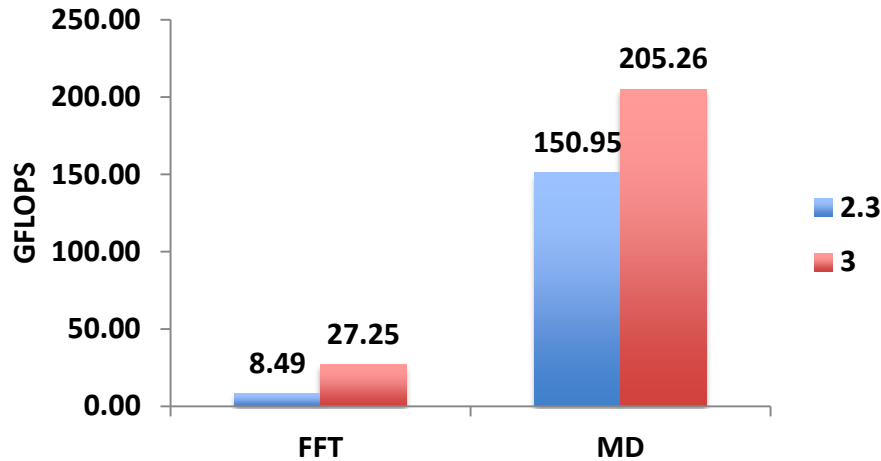
- Motivation
  - Extremely common scientific kernel
  - Bandwidth bound, and much harder to get performance than GEMM
- Basic design
  - 3 Algorithms, padded & unpadded data
  - CSR and ELLPACKR data formats
  - Supports random matrices or matrix market format
  - Example: Gould, Hu, & Scott: expanded system-3D PDE.



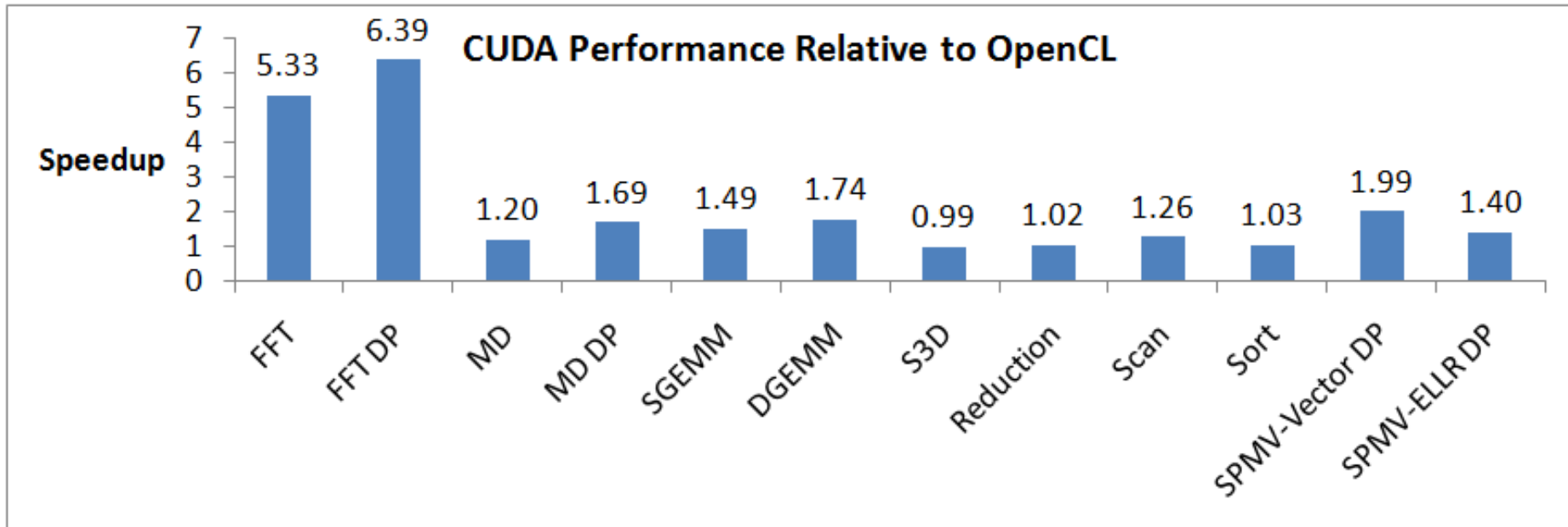
# SpMV Performance



# Comparing CUDA and OpenCL



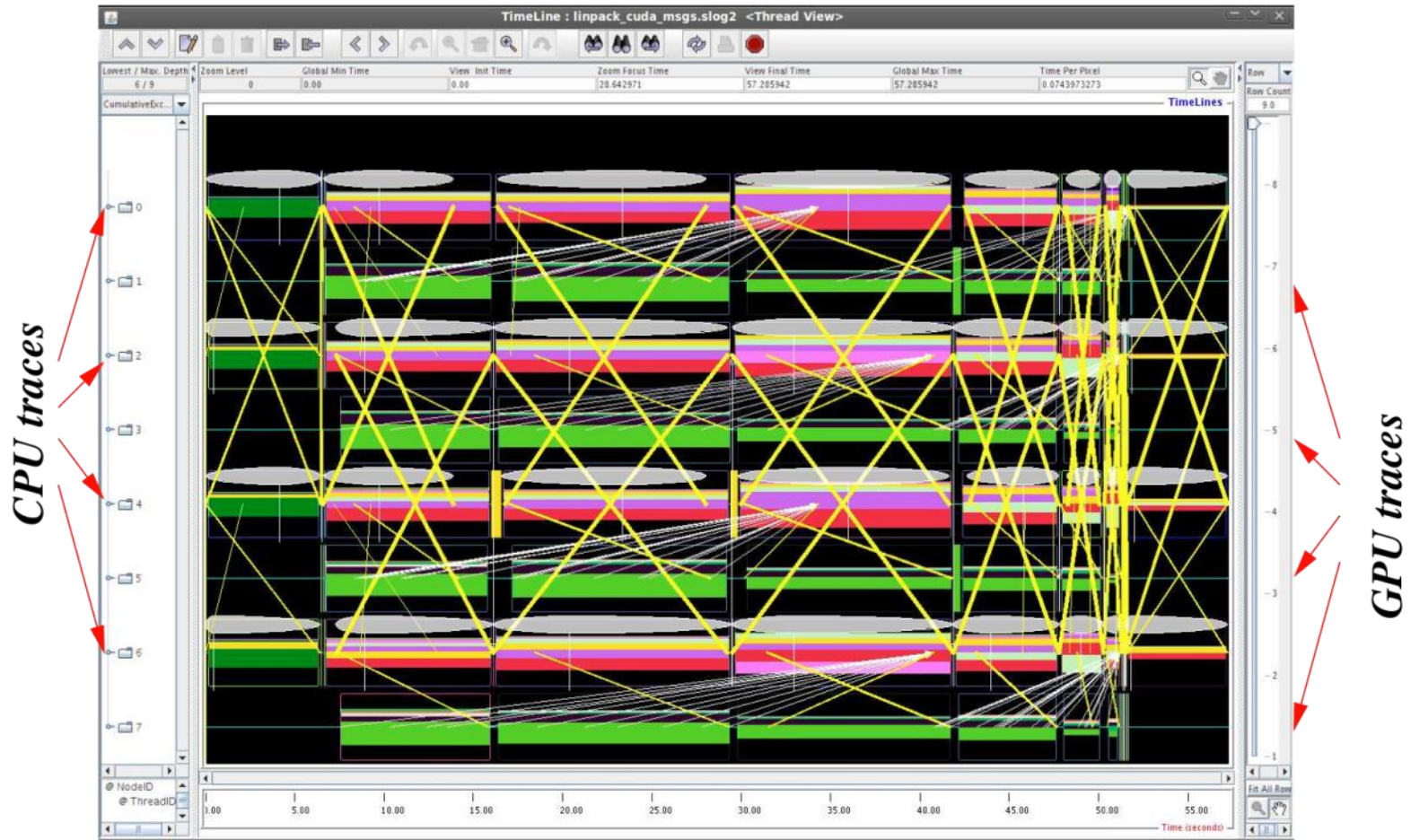
Single precision, Tesla C1060 GPU  
Comparing NVIDIA OpenCL  
implementation from 2.3 and 3.0 GPU  
Computing SDK



# Performance and Correctness



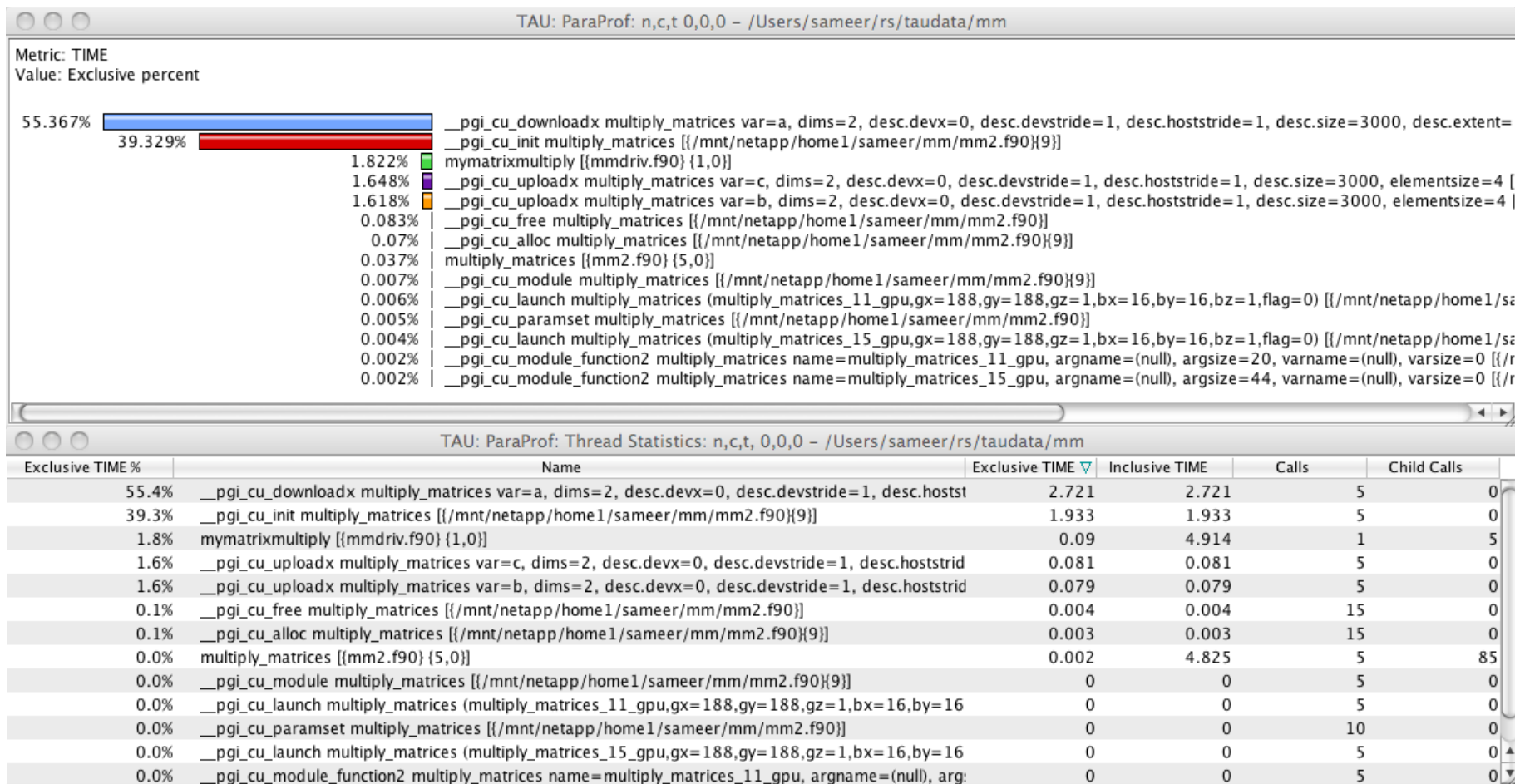
# Vancouver: Integrated Performance Analysis of MPI/GPU Applications



CUDA memory transfer (white)

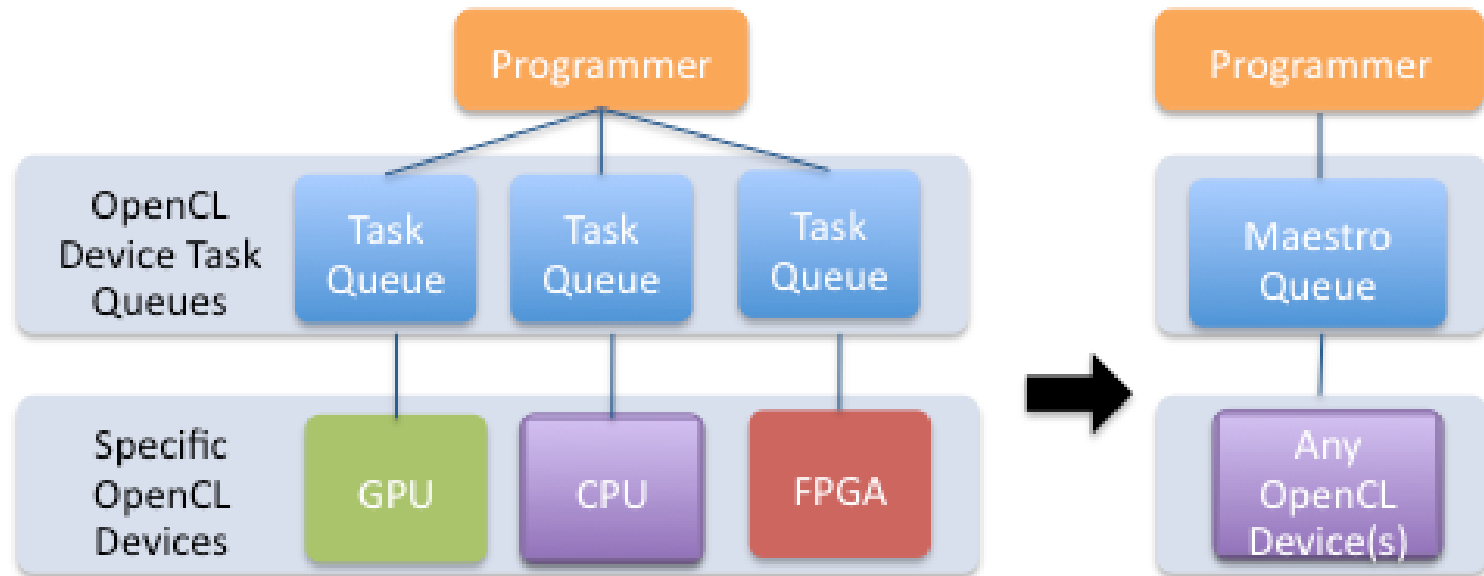
MPI communication (yellow)

# Vancouver: Integrated Performance Analysis of Compiler CUDA Generated Applications



# Performance Portability

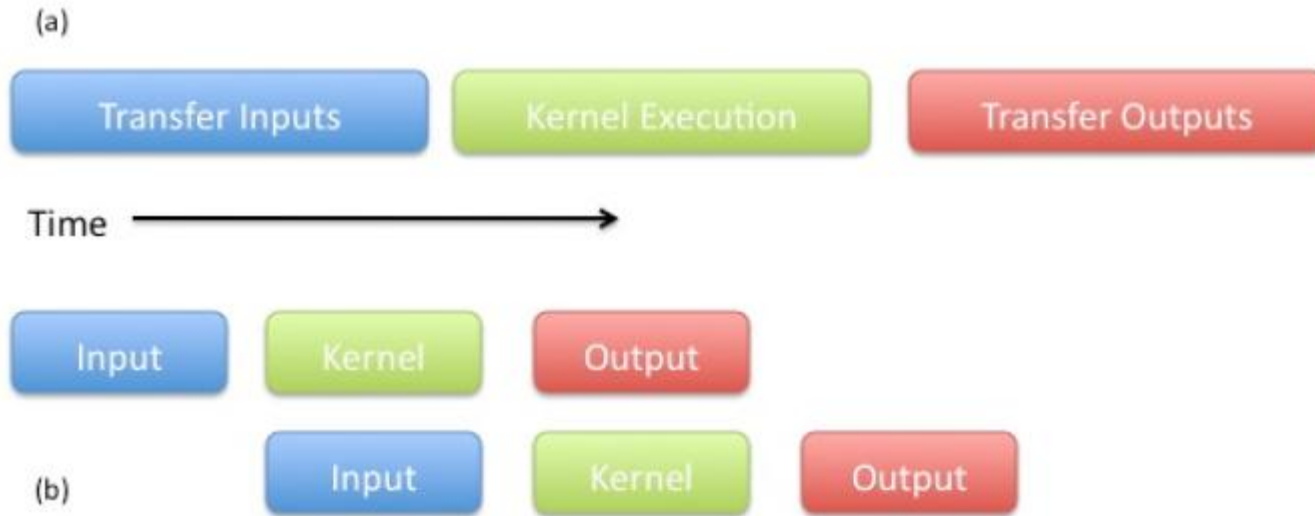
# Maestro



- Portability
- Load balancing
- Autotuning

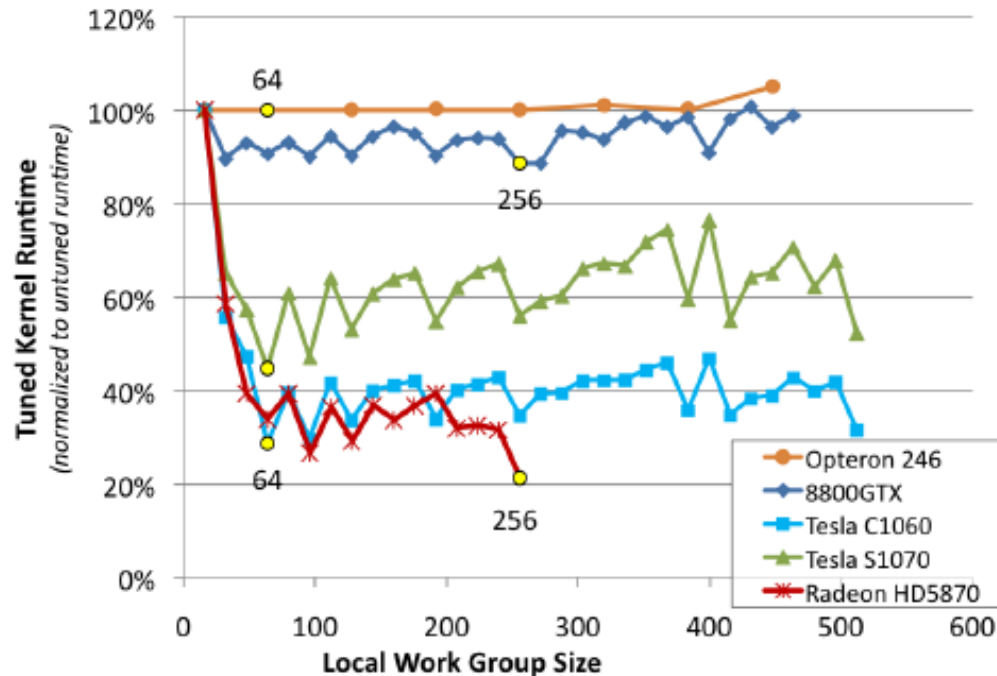
K. Spafford, J. Meredith, and J. Vetter, "Maestro: Data Orchestration and Tuning for OpenCL Devices," in *Euro-Par 2010 - Parallel Processing*, vol. 6272, Lecture Notes in Computer Science, P. D'Ambra, M. Guarracino et al., Eds.: Springer Berlin / Heidelberg, 2010, pp. 275-86.

# Maestro: Multibuffering



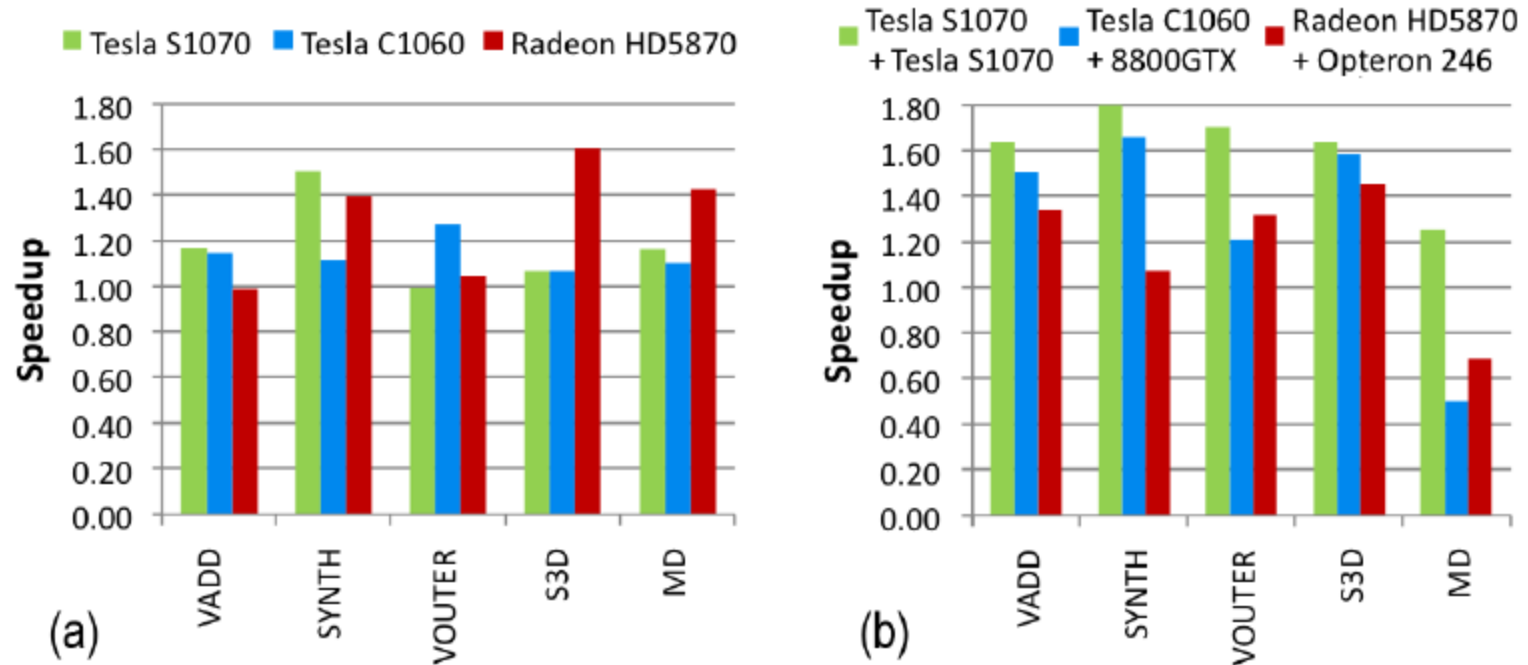
**Fig. 2. Double Buffering**—This figure contrasts the difference between (a) the function offload model and (b) a very simple case of double buffering. Devices which can concurrently execute kernels and transfer data are able to hide some communication time with computation.

# Maestro : Autotuning Workgroups



**Fig. 3. Autotuning the local work group size** – This figure shows the performance of the MD kernel on various platforms at different local work group sizes, normalized to the performance at a group size of 16. Lower runtimes are better.

# Combined Autotuning Results



**Fig. 6. Combined autotuning results** – (a) Shows the combined benefit of autotuning both the local work group size the double buffering chunk size for a single GPU of the test platforms. (b) Shows the combined benefit of autotuning both the local work group size and the multi-GPU load imbalance using both devices (GPU+GPU or GPU+CPU) of the test platforms. Longer bars are better.

# **#2: Memory Bandwidth and Capacity**



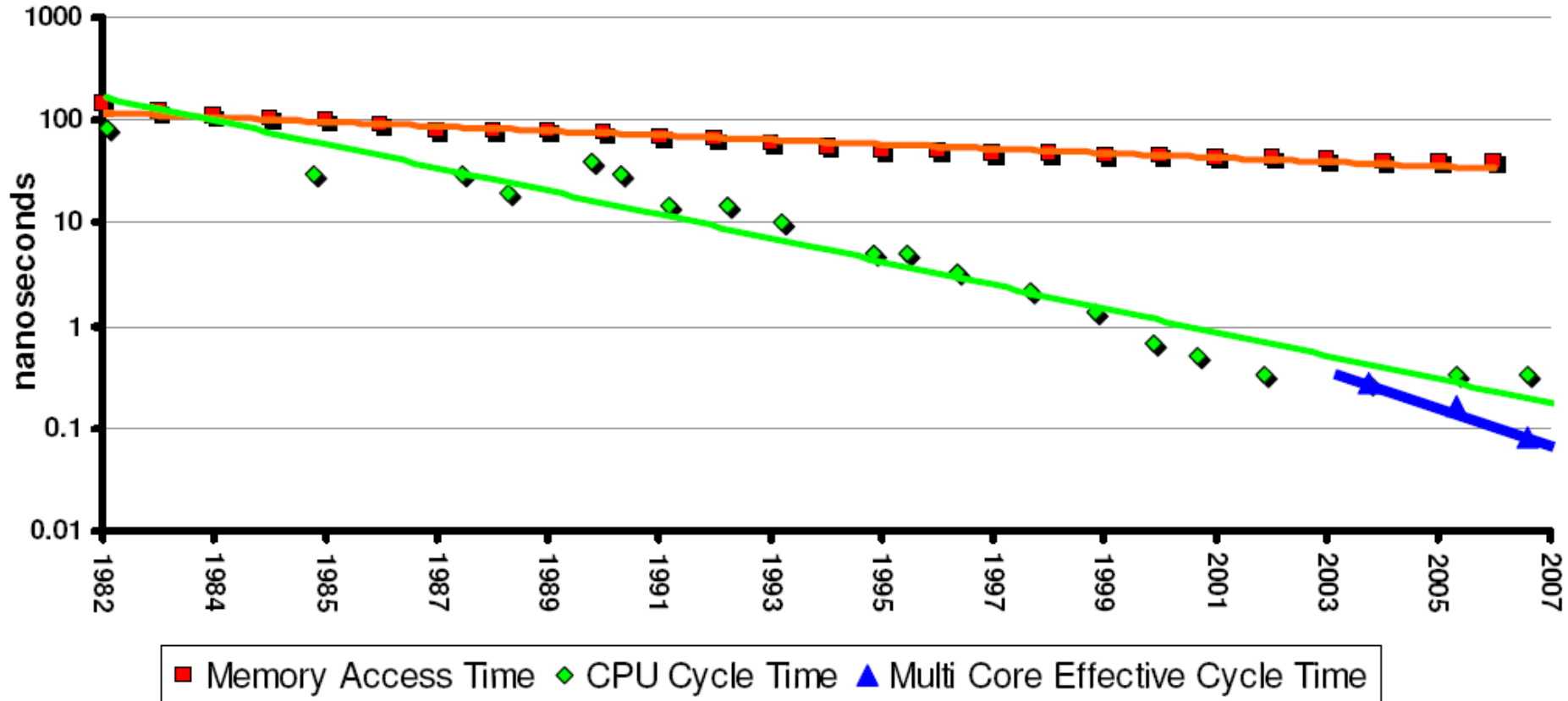
# Critical Concern : Memory Capacity

	2010	2018	Factor Change
System peak	2 Pf/s	1 Ef/s	500
Power	6 MW	20 MW	3
System Memory	0.3 PB	10 PB	33
Node Performance	0.125 Tf/s	10 Tf/s	80
Node Memory BW	25 GB/s	400 GB/s	16
Node Concurrency	12 CPUs	1,000 CPUs	83
Interconnect BW	1.5 GB/s	50 GB/s	33
System Size (nodes)	20 K nodes	1 M nodes	50
Total Concurrency	225 K	1 B	4,444
Storage	15 PB	300 PB	20
Input/Output bandwidth	0.2 TB/s	20 TB/s	100

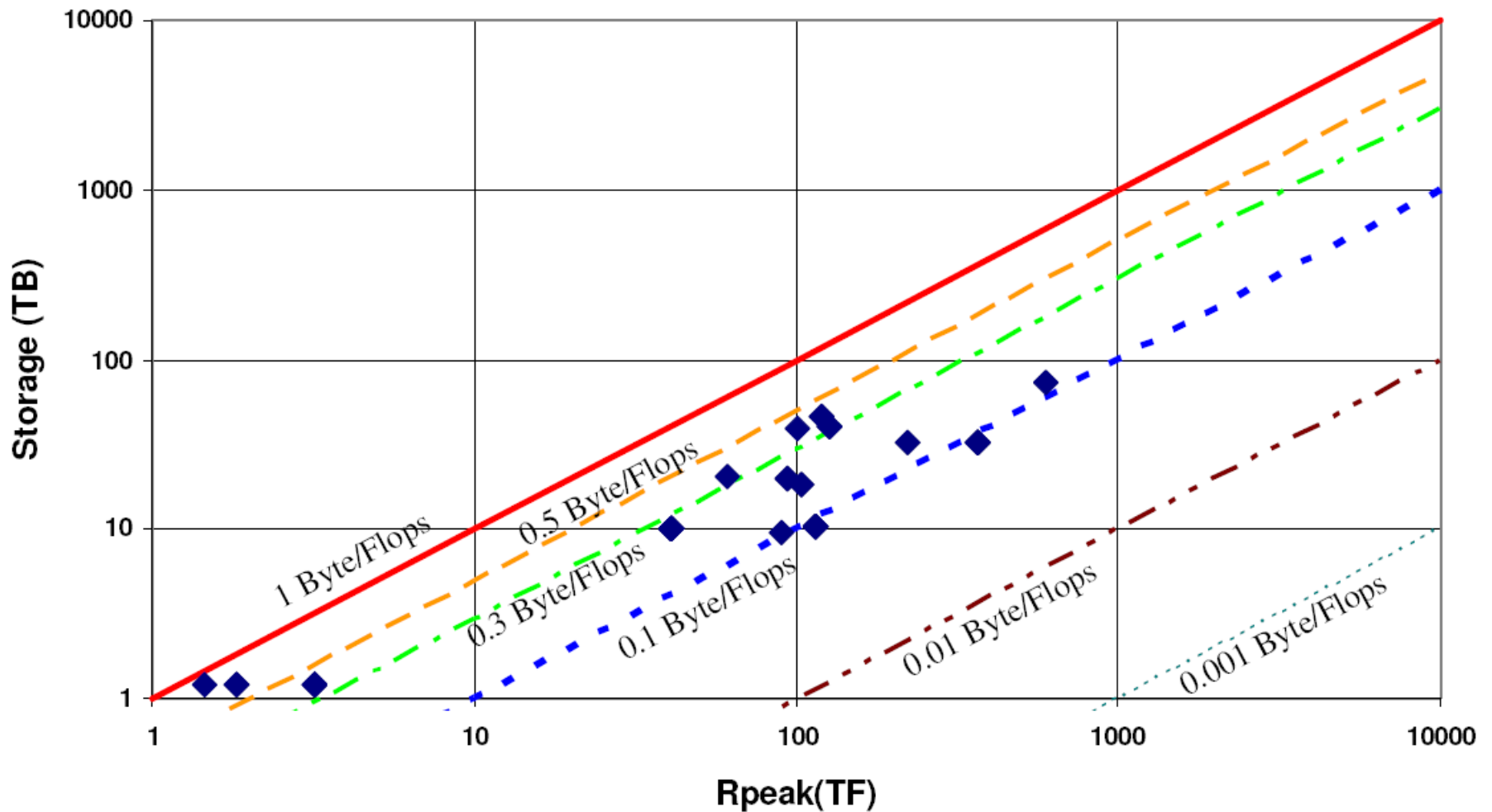
**Table 1:** Potential Exascale Computer Design for 2018 and its relationship to current HPC designs. <sup>58</sup>

- Small memory capacity has profound impact on other features
- Feeding the core
- Poor efficiencies
- Small messages, I/O

# Memory Performance



# Memory Capacity



# New Technologies Offer Promise

Device Type	HDD	DRAM	NAND Flash	FRAM	MRAM	STTRAM	PCRAM	NRAM
Maturity	Product	Product	Product	Product	Product	Prototype	Product	Prototype
Present Density	400Gb/in <sup>2</sup> [7]	8Gb/chip [9]	64Gb/chip [10]	128Mb/chip	32Mb/chip	2Mb/chip	512Mb/chip	NA
Cell Size (SLC)	(2/3)F <sup>2</sup>	6F <sup>2</sup>	4F <sup>2</sup>	6F <sup>2</sup>	20F <sup>2</sup>	4F <sup>2</sup>	5F <sup>2</sup>	5F <sup>2</sup>
MLC Capability	No	No	4bits/cell	No	2bits/cell	4bits/cell	4bits/cell	No
Program Energy/bit	NA	2pJ	10nJ	2pJ	120pJ	0.02pJ	100pJ	10pJ [11]
Access Time (W/R)	9.5/8.5ms [8]	10/10ns	200/25us	50/75ns	12/12ns	10/10ns	100/20ns	10/10ns [11]
Endurance/Retention	NA	10 <sup>16</sup> /64ms	10 <sup>5</sup> /10yr	10 <sup>15</sup> /10yr	10 <sup>16</sup> /10yr	10 <sup>16</sup> /10yr	10 <sup>5</sup> /10yr	10 <sup>16</sup> /10yr

Device Type	RRAM	CBRAM	SEM	Polymer	Molecular	Racetrack	Holographic	Probe
Maturity	Research	Prototype	Prototype	Research	Research	Research	Product	Prototype
Present Density	64Kb/chip	2Mb/chip	128Mb/chip	128b/chip	160Kb/chip	NA	515Gb/in <sup>2</sup>	1Tb/in <sup>2</sup>
Cell Size	6F <sup>2</sup>	6F <sup>2</sup>	4F <sup>2</sup>	6F <sup>2</sup>	6F <sup>2</sup>	N/A	N/A	N/A
MLC Capability	2bits/cell	2bits/cell	No	2bits/cell	No	12bits/cell	N/A	N/A
Program Energy/bit	2pJ	2pJ	13pJ	NA	NA	2pJ	N/A	100pJ [12]
Access Time (W/R)	10/20ns	50/50ns	100/20ns	30/30ns	20/20ns	10/10ns	3.1/5.4ms	10/10us
Endurance/Retention	10 <sup>8</sup> /10yr	10 <sup>8</sup> /Months	10 <sup>9</sup> /days	10 <sup>4</sup> /Months	10 <sup>5</sup> /Months	10 <sup>16</sup> /10yr	10 <sup>5</sup> /50yr	10 <sup>5</sup> /NA

# Blackcomb: Hardware-Software Co-design for Non-Volatile Memory in Exascale Systems

A comparison of various memory technologies

	SRAM	DRAM	NAND Flash	PC-RAM	STT-RAM	R-RAM
Data Retention	N	N	Y	Y	Y	Y
Memory Cell Factor (F <sup>2</sup> )	50-120	6-10	2-5	6-12	4-20	<1
Read Time (ns)	1	30	50	20-50	2-20	<50
Write / Erase Time (ns)	1	50	10 <sup>6</sup> -10 <sup>8</sup>	50-120	2-20	<100
Number of Rewrites	10 <sup>16</sup>	10 <sup>16</sup>	10 <sup>5</sup>	10 <sup>10</sup>	10 <sup>15</sup>	10 <sup>15</sup>
Power Read/ Write	Low	Low	High	Low	Low	Low
Power (Other than R/W)	Leakage Current	Refresh Power	None	None	None	None

## Novel Ideas

- **New resilience-aware designs for non-volatile memory applications**
  - Mechanical-disk-based data-stores are completely replaced with energy-efficient non-volatile memories (NVM).
  - Most levels of the hierarchy, including DRAM and last levels of SRAM cache, are completely eliminated.
- **New energy-aware systems/applications for non-volatile memories (nanostores)**
  - Compute capacity, comprised of balanced low-power simple cores, is co-located with the data store.

## Impact and Champions

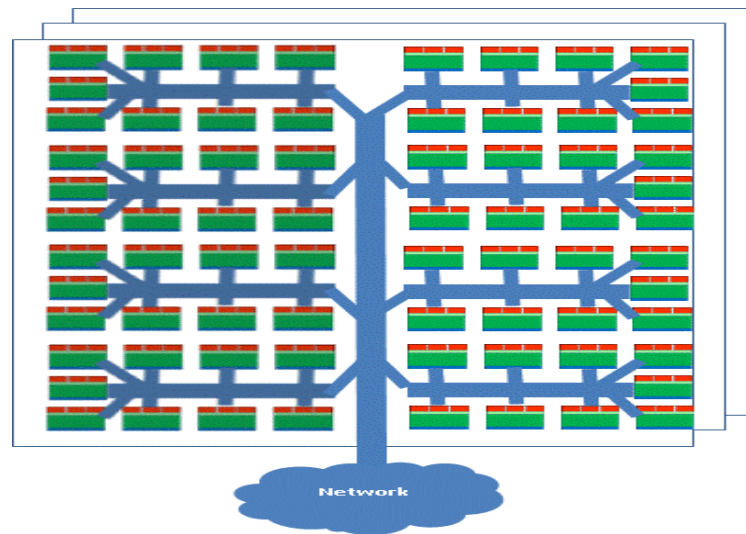
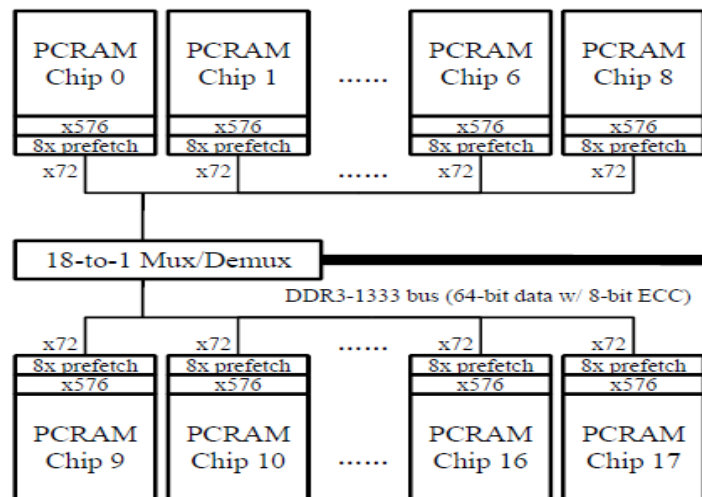
- **Reliance on NVM addresses device scalability, energy efficiency and reliability concerns associated with DRAM**
  - **More memory** – NVM scalability and density permits significantly more memory/core than projected by current Exascale estimates.
  - **Less power** – NVMs require zero stand-by power.
  - **More reliable** – alleviates increasing DRAM soft-error rate problem.
- **Node architecture with persistent storage near processing elements enables new computation paradigms**
  - Low-cost checkpoints, easing checkpoint frequency concerns.
  - Inter-process data sharing, easing in-situ analysis (UQ, Visualization)

## Milestones

- Identify and evaluate the most promising non-volatile memory (NVM) device technologies.
- Explore assembly of NVM technologies into a storage and memory stack
- Build the abstractions and interfaces that allow software to exploit NVM to its best advantage
- Propose an exascale HPC system architecture that builds on our new memory architecture
- Characterize key DOE applications and investigate how they can benefit from these new technologies

# Opportunities go far beyond a plugin replacement for disk drives...

- New distributed computer architectures that address exascale resilience, energy, and performance requirements
  - replace mechanical-disk-based data-stores with energy-efficient non-volatile memories
  - explore opportunities for NVM memory, from plug-compatible replacement (like the NV DIMM, below) to radical, new data-centric compute hierarchy (nanostores)
  - place low power compute cores close to the data store
  - reduce number of levels in the memory hierarchy
- Adapt existing software systems to exploit this new capabilities

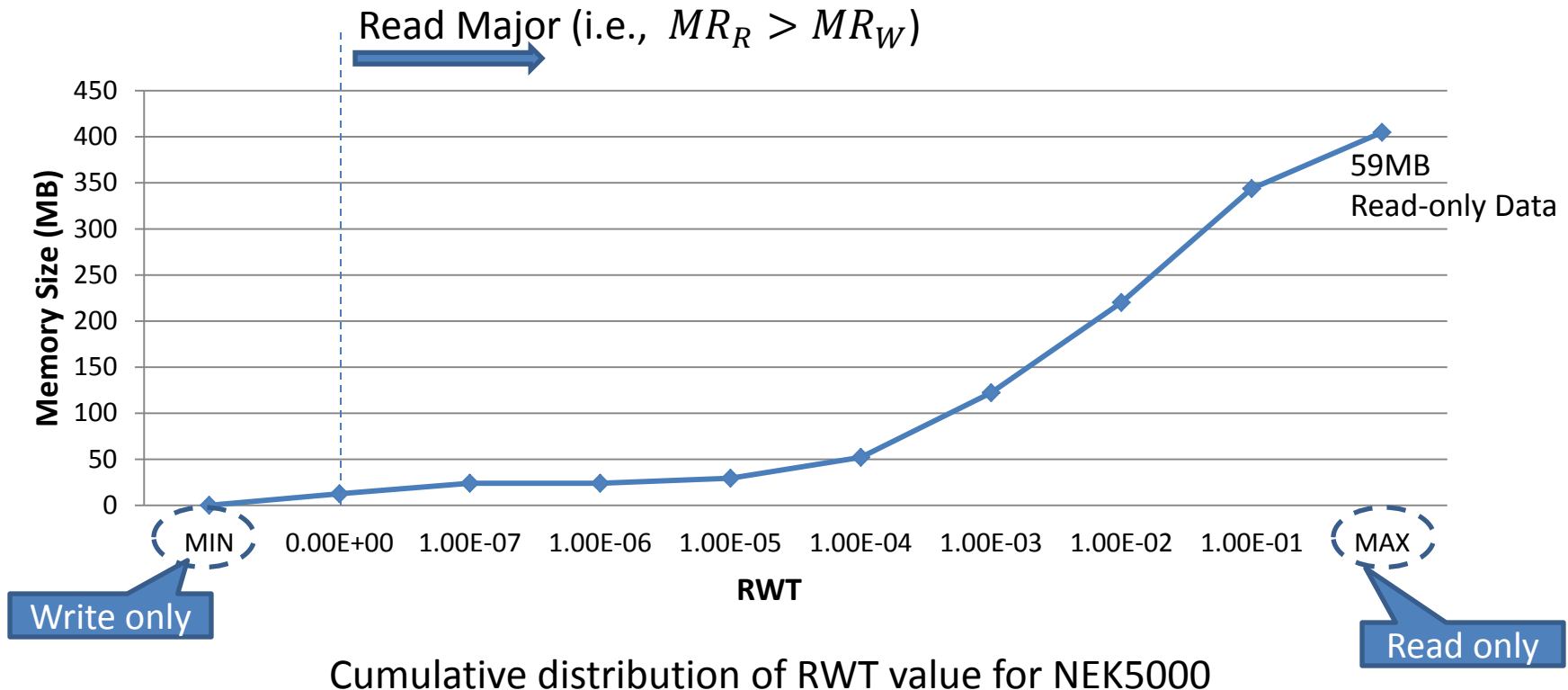


# How might these new technologies help applications in new ways?

- Major differences
  - Persistence
  - Read/Write asymmetries
  - Zero standby power
- Assuming new (disruptive) technologies, how should applications use them in an exascale architecture?
  - What is the proper level of integration into the memory hierarchy?
  - What data should be stored in this memory?
  - How should we change the usage scenarios?

# App1

- A computational fluid dynamic solver
  - cover a broad range of applications
  - We instrument the eddy problem, a 2D problem
- RWT





# Summary

- Toward Exascale
  - Highlights from recent projections for exascale
- Challenges
  - Micro, macro power
  - Memory capacity and bandwidth
  - Parallelism
  - Programmability
  - Algorithms
- Research solutions
  - Heterogeneity with GPUs: Keeneland
  - Programming models: SHOC, Maestro
  - NVRAM
- Applications teams should
  - Prepare for hardware diversity!
  - Think of new ways to employ technologies
- <http://ft.ornl.gov>